PMA Prozeß- und Maschinen-Automation GmbH

Industrial controller KS 94

KS94 INTERBUS

INTERBUS

Interface description INTERBUS

9499 040 56911

Valid from: 8363

® is a registered trademark of the
INTERBUS user organization

DAC ® is a patented method and a registered trademark
of Regeltechnik Kornwestheim GmbH

© PMA Prozeß- und Maschinen-Automation GmbH 1999 Printed in Germany (9908)
All rights reserved. No part of this document may be reproduced or
published in any form or by any means without prior written permission
from the copyright owner.

A publication of PMA Prozeß- und Maschinen Automation
P.O. Box 310229
D-34058 Kassel
Germany

Inhalt

1. General 5
1.1 INTERBUS interface 5
1.2 User advantages 5
1.3 Scope of delivery 6
2. Hints for operation 7
2.1 Operation. 7
2.2 Remote/Local 7
2.2.1 Local switch-over via interface 7
2.3 Interface connection 8
2.3.1 Laying cables 8
2.4 INTERBUS diagnosis 8
2.4.1 Display signification 9
3. Special functions 10
3.1 'Back-Up' controller operation. 10
3.1.1 Forcing 10
3.1.2 Controller output locking. 10
4. Processdata 11
4.1 Projecting 11
4.2 Process data transmission 11
4.2.1 ***Controller adjustments*** 11
4.2.2 ***INTERBUS network.*** ***12***
4.2.3 Introducing KS94 instrument logos in CMD. 13
4.3 Process data modules 14
4.3.1 Statuswords 15
4.3.2 Controlwords 16
4.3.3 Locking mechanism with changes 16
4.4 Data_types 17
4.4.1 Mapping on INTERBUS process data channel 17
4.4.2 Mapping on INTERBUS parameter channel 17
5. Example projects for CMD 18
5.1 Example loading procedure 19

6. Parameter transmission (PCP) 20
6.1 Supported PCP services: 20
6.2 Communication principle for operation via PCP 21
6.2.1 Description of the most important parameters 21
6.2.2 Building up a communication 22
6.2.3 Sending data to the bus unit 22
6.2.4 Reading data from a bus unit 22
6.2.5 Terminating the communication 22
6.3 KS94-IBS objects 23
6.3.1 Survey 23
6.3.2 Transmission parameters 23
6.3.3 Object 2000_{hex} 'MEM_RD_PAR' 24
6.3.4 Object 2001_{hex} 'MEM_WR_PAR' 24
6.3.5 Object 2002_{hex} 'SINGLE_AC' 25
6.3.6 Object 2003_{hex} 'BLOCK_AC' 26
6.3.7 Object 2004_{hex} 'TEXT_AC' 27
6.3.8 Object 2005_{hex} 'DECADE_AC' 28
6.4 PCP error messages 29
7. Function block protocol 30
7.1 **Data structuring** 30
7.2 CODE tables 31
7.2.1 Configuration words (C.xxxx) 31
7.2.2 INSTRUMENT (FB no.: 0 type no.: 0) 32
7.2.3 INPUT (FB no.: 61 Type no.: 110) 40
7.2.4 CONTR (FB no.: 50 Type no.: 90) 44
7.2.5 ALARM (FB no.: 51 type no.: 45) 52
7.2.6 OUTPUT (FB no.: 81 Type no.: 111) 53
8. Annex 54
8.1 **Legend of terms and abbreviations** 54
9. Index 55

1 General

The (9407-92x-4xxx1) versions of Industrial Controller KS94 are provided with an INTERBUS – remote bus interface for transmission of the process and parameter data. Connection is on the instrument rear. The serial communication interface permits communication with supervisory systems, visualization tools, etc.

Another interface, which is always provided as standard is the front-panel PC interface. It is used for connection of the engineering tool, which runs on a PC.

1.1 INTERBUS interface

INTERBUS communication is according to the master/slave principle. KS94/IBS is always slave.

Lead as well as physical and electrical properties of the interface are as follows:

- Network topology
  Active ring with separate leads for both communication directions in one cable.
  Stub lines via special components such as bus terminals or bus coupling units are possible.
- Cable definition according to EN 50254-2-2:
  - Cable: Cu-strand conductor, 3 twisted pairs
  - Characteristic impedance: 80 -120 Ω (typ. 100 Ω)
  - Distributed capacity: < 60 pF/m
  - Loop resistance: < 250 Ω/km
  - Conductor cross-section: > 0,20 mm^2
- Baudrates and cable lengths
  According to EN 50254-2-2, cables with a max. length of 400 m (Cu) are permissible between two field bus units. The transmission rate is fixed to 500 kbit/s.
- BaudrateInterface
  RS485 with sub-D-connector/socket; on-site assembly possible
- Max. number of stations in a ring: 256 (max. 85 KS94/IBS).

1.2 User advantages

KS94 with INTERBUS interface offers many advantages related to handling and integration into an INTERBUS network.

- Diagnosis and monitoring
  Display of bus messages
  - Plain text front panel display
  - Diagnosis LEDs on the rear
  - Diagnosis information selectable for output via front-panel LED

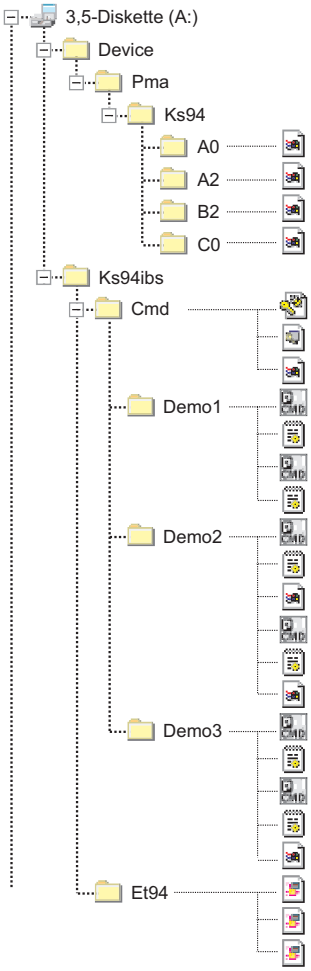
  Bus error signalling via e.g. relays

- Special features
  Configurable process data modules
  Direct input and output reading and writing
  Input forcing
  Output disabling
  Back-up controller function
  Simple connection even to small PLCs

1.3 Scope of delivery

The engineering set comprises:

- Floppy



A0.dsh	Description of unit structure A.0 ID: 51
A2.dsh	Description of unit structure A.2 ID: 240
B2.dsh	Description of unit structure B.2 ID: 240
C0.dsh	Description of unit structure C.0 ID: 240
Exdevdb.ldb	Database for external unit description
Exdevdb.mdb	
Pma.icl	KS94 instrument logo for CMD
Ks94dmo1a.bg4	Demonstration project 1a for CMD
Ks94dmo1a.ini	
Ks94dmo1b.bg4	Demonstration project 1b for CMD
Ks94dmo1b.ini	
Ks94dmo2a.bg4	Demonstration project 2a for CMD
Ks94dmo2a.ini	
Ks94dmo2a.zwt	
Ks94dmo2b.bg4	Demonstration project 2b for CMD
Ks94dmo2b.ini	
Ks94dmo2b.zwt	
Ibsdmo1.e94	'ET/KS94' engineering file for Demo1
Ibsdmo2.e94	'ET/KS94' engineering file for Demo2

- Interface description for INTERBUS

2 Hints for operation

2.1 Operation

KS94 data can be read or displayed and changed from the front-panel PC interface_PC interface or via the serial interface.

After delivery of controller KS94, the PC interface is active. Controller configuration and parameter setting are supposed to be done by means of the engineering tool before commissioning.

The interface;Switching over to front-panel interface;Switching over to busSwitching over to the serial interface is either

- via operator dialogue (front panel):

  Press ⏎ ≥ during 3 sec. → Para blinks
  Press ▲ , until CBus blinks → Press ⏎ shortly.

Display	≙	Switching over to rear interface
CBus		

- or by activating 'REMOTE' (→ page 7). Returning to LOCAL does not cause switch-over to the front-panel interface.

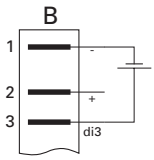
Switch-over to the PC-interface is only possible with the unit set to LOCAL.

- Via operator dialogue (front panel):

  Press ⏎ ≥ during 3 sec. → Para blinks
  Press ▲ until CFrnt blinks. → Confirm ⏎ shortly.

Display	≙	Switching over to front-panel interface
CFrnt		

2.2 Remote/Local



Remote/Local with serial interface are provided with a hardware input (di3) for switching over between REMOTE and LOCAL operation (R/L).

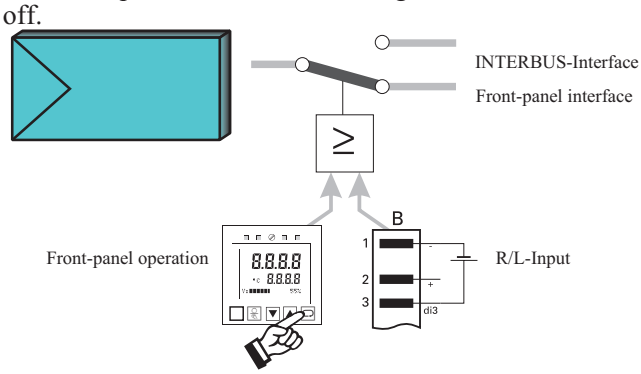
During 'REMOTE' mode, all operations via the serial interface (writing and reading) are permissible. The following operations are still possible via the keys of the local operating panel:

- Display switch-over
- Parameter display/reading, however, without changing
- Configuration data display/reading, however, without changing

During remote mode, operation of the PC interface is not possible. When switching over from LOCAL to REMOTE, any active PC interface is switched off.

During the 'LOCAL' mode, all data can only be read via the serial interface.

Modifications are not possible, except:
any data related only to the interface,
or which are not adjustable
via local operation.



2.2.1 *Local switch-over via interface*

Local switch-over via the field bus interface is possible. Conditions:

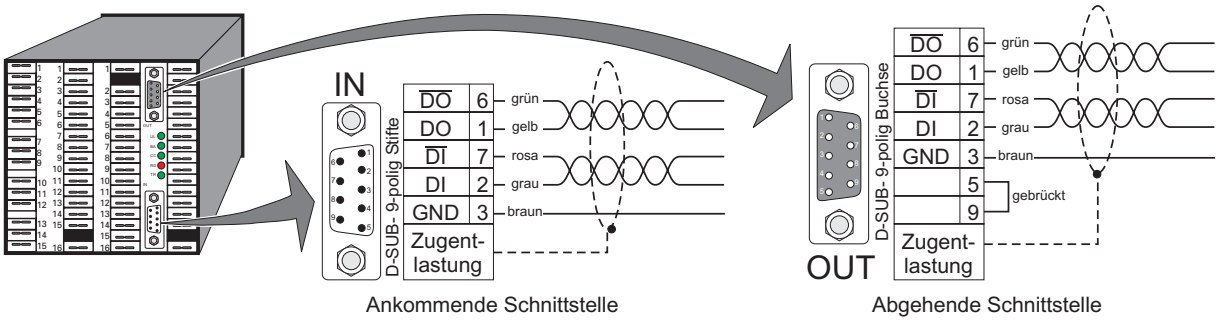
- Unit is in REMOTE mode (contact di3 closed).
- Switch-over to local mode is possible via interface command. Inverse switch- over (return to remote) is also possible.
- default setting is the remote mode.
- During HW mode local, this setting is ineffective.
- During transmission to HW mode local or during power ON, the default value is adjusted.

⚠ Caution! Returning to remote mode is not possible when switching over to the front-panel interface. With INTERBUS, all cyclic write data must be rewritten when returning from Local to Remote, in order to ensure the busmaster status defined last.

2.3 Interface connection

Connect the INTERBUS to the 9-pole sub-D connector on the controller rear (→ Fig.:1) and tighten the mounting screws manually.

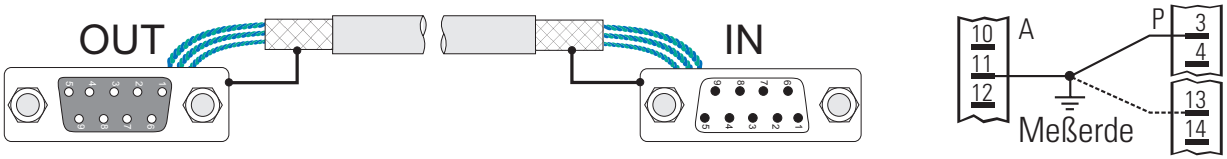
Fig.: 1 INTERBUS connection



Appropriate cables must be assembled by the user. For this, the general cable specifications_cable specifications to EN 50254-2-2 must be followed. Signal pairs (e.g. DO / DO) must be taken via twisted wire pairs. Metal connector housings should always be used.

Connect the bus cable screening as shown in Fig.: 2 . To activate the screening, the measurement earth must be connected accordingly at the controller.

Fig.: 2 Screening connection



2.3.1 Laying cables

When laying the cables, the general hints for laying cables given by the master module supplier must be taken into account:

- Cable run inside buildings (inside and outside cabinets)
- Cable run outside buildings
- Potential compensation
- Cable screening
- Measures against interference voltages
- Connection of the measurement earth at KS94

2.4 INTERBUS diagnosis

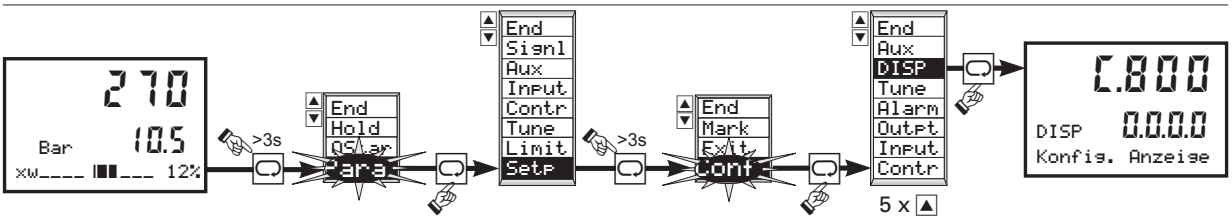
The LEDs to be used for INTERBUS diagnosis are located on the instrument rear. The signification of LEDs is given in the following table:

UL	LED - Description	LED - colour	LED = on	LED = off
BA	UL (supply voltage)	green	Voltage provided, no reset cycle	Reset active or no voltage provided
CC	BA (Bus aktive)	green	Bus active	Bus stopped
RD	CC [1] (Cabel Check)	green	Incoming remote bus active	Incoming remote bus switched off
TR	RD (Remote Disable)	red	Outgoing bus switched off	Outgoing bus active
	TR (Transmit/Receive)	green	PCP - communikation active	no PCP - communication

Additionally, there are two possibilities of bus message display_bus status display, which can be adjusted during controller configuration (C.800):

- Plain text message on the lower display line
  C.800; UsrTx = 1 or 2 (user text via control input or function statuses)
- Indication via LED (front-panel LEDs)
  C.800; LED = 4 (INTERBUS error messages)
  Allocation of front-panel LEDs and INTERBUS diagnosis LEDs on the controller rear:
  UL = LED 1 (left LED); BA = LED 2; CC = LED 3; RD = LED 4 (right LED)

Fig.: 3 Selection of configuration word C.800



Now, the relevant configuration can be selected by pressing the ⮌ key shortly.

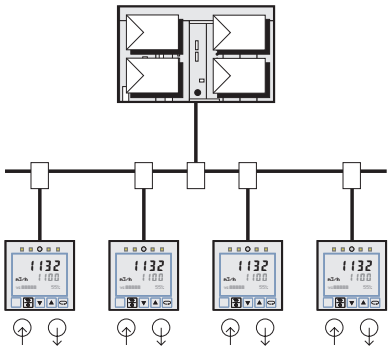
2.4.1 Display signification

Plain text display	UL	BA	CC	RD	TR	Signification
IBS:Mod.unknown	off	d.c.	d.c.	d.c.	d.c.	Module error, interface_interface
IBS:no conn	on	off	off	on	d.c.	No connection to connecting module, cable error
IBS:Bus ready	on	off	on	on	d.c.	Bus ready status (RDY), connection ok., no data exchange
keine Anzeige	on	on	on	off	off	Run status (RUN), data exchange
IBS:PCP aktive	on	on	on	off	on	Run status and PCP channel set up

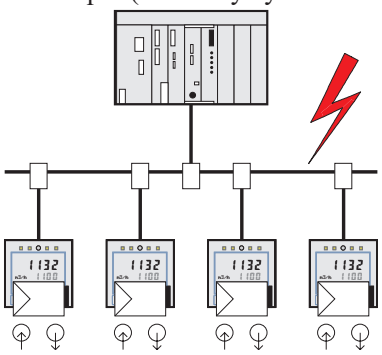
1) Incoming bus monitoring; if the incoming cable is o.k. and the connected module is not in Reset, CC is active.

3 Special functions

3.1 'Back-Up' controller operation



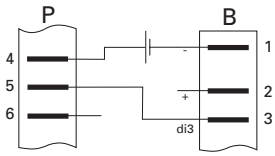
'Back-Up' controller operation. During normal operation, calculation of controller outputs is in the PLC. The controllers are used for process value measurement and correcting value output (incl. duty cycle conversion and display).



In case of trouble, i.e. with failure of PLC or bus communication, control operation is continued independently and bumplessly by the KS 94 controllers.

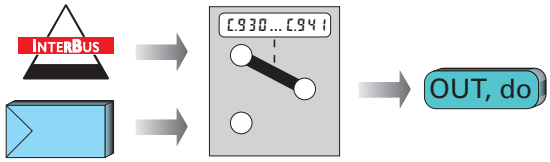
Determination that the controller switches to automatic mode automatically when changing from Remote to Local is by configuration. Selection 'back-up operation' is in configuration word C.191 (S A/M = 4). If the application requires that the controller goes to automatic mode, i.e. that the control operation is continued by the controller in case of INTERBUS communication error, the following measures must be taken:

- Definition of an alarm dependent of INTERBUS status (e.g. C.600 ...; Src = 23; Fnc = 6)
- Output of this alarm on a relay or logic output
- External wiring of this alarm to the remote/local input (di3)



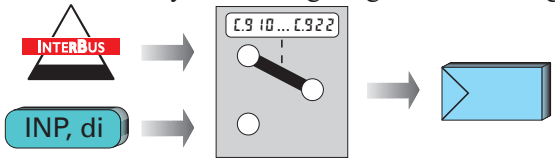
3.1.1 *Forcing*

Analog forcing inputs can be determined for the controller via INTERBUS after configuring them accordingly (C.910, C.911), also if they are not existing physically. Digital inputs can also be set



(C.920 ... C.922).

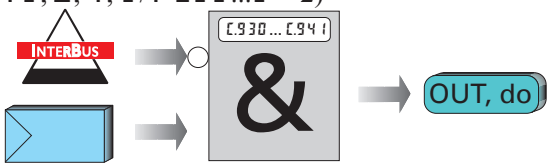
Analog outputs can be written directly after configuring them accordingly (C.930, C.931). Digital outputs



can also be set directly (C.940, C.941).

3.1.2 *Controller output locking*

Via INTERBUS, the KS94 digital outputs can be locked.
(C.930, C.941; FOUT1, 2, 4, 5/Fdo1 ...6 = 2)



FOUT1 can be locked only, if it was configured as relay or logic.

4 Processdata

During data transmission, distinction between cyclically transmitted process data and acyclically transmitted parameter /configuration data is made. The I/O data field structure is modular for matching it to the requirements of the control task.
The process data to be transmitted can be determined by engineering (see section 4.2 page11).

Process data are transmitted as integer16, type “FixPoint1” is determined as value range. FixPoint1 has one fixed digit behind the decimal point, e.g. 30°C is transmitted as 3000. The permissible range is -3000,0 to +3200,0.

4.1 Projecting

For integrating a KS94 into an INTERBUS network, preparative measures must be taken.

- Adjust the controller by means of Engineering Tool ”ET/KS94".
- Integrate the controller into an INTERBUS network.

4.2 Process data transmission

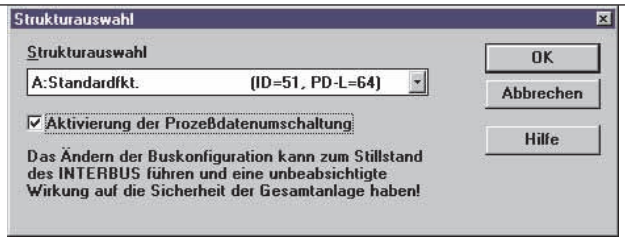
Process data are read and written cyclically by the controller. The minimum poll time of 200ms is met, unless an access via the parameter channel is made simultaneously.
Output data sent to KS94 are compared with the values sent previously and only processed with deviation. If one of the values is faulty, bit 11 in status word 1 is set, until there are no faulty values any more.

4.2.1 Controller adjustments

In addition to application-specific controller adjustment (configuration and parameter setting), the INTERBUS functions must be adjusted.

- To meet various memory space and transmission time requirements, selection between several process data modules is possible (→ see section). Standard function A.0 (ID = 51; PD-L = 64) is adjusted by default. Selection of the process data module is via
  change→configuration→additionals→structure selection.

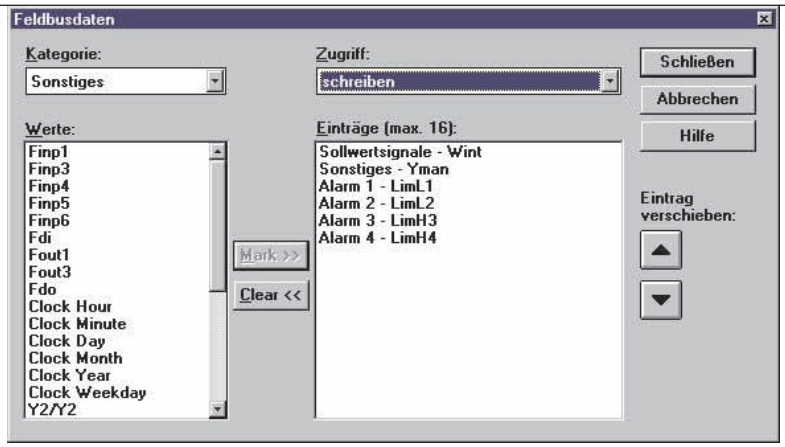
Fig.: 4 Process data module selection



With the checkbox for "activation of process data switch-over" activated, the KS94 Interbus connecting structure is changed directly. In this case, corresponding measures can be taken directly by the master (switch-over to ‘Alarm-Stop’ and then to ‘Run’). Unless this function is activated, structure changing is only active after ‘power-on’.

- The user can select any data to be transmitted cyclically from all instrument process data and instrument parameters by means of engineering tool 'ET/KS94', i.e. he can compose his transmission data himself.
  The data for cyclic transmission are selected via
  change → parameter → miscellaneous → fieldbusdata .

Fig.: 5 Controller data allocation for the field bus

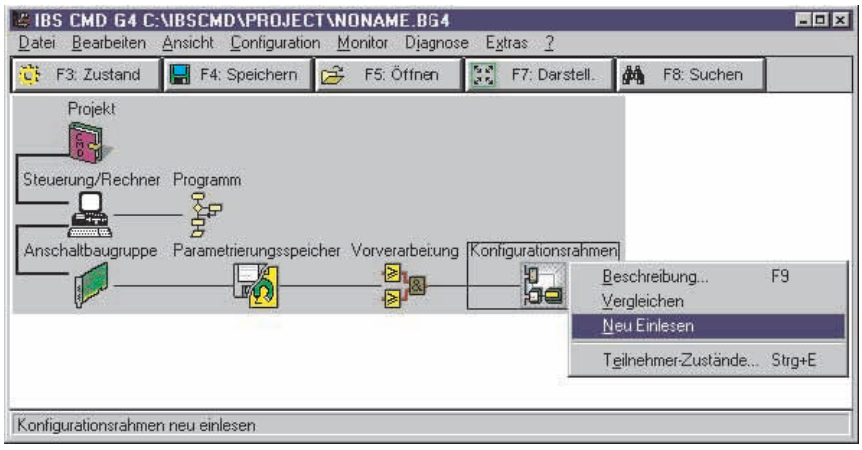


Max. 16 data for reading and 16 data for writing can be selected. Dependent of process data module, the first two data (structure A), the first six data (structure B) or all selected data (structure C) are used.

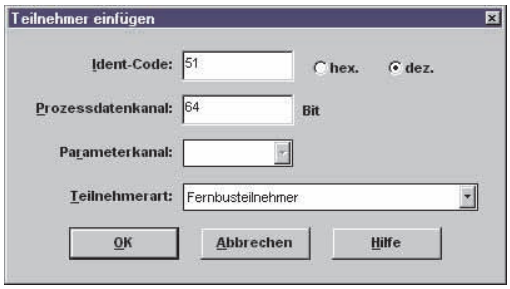
4.2.2 INTERBUS network

For KS94 operation, the adjusted ident code and the used process data length must be reported to the INTERBUS. With a Phoenix Contact CMD tool, this is possible by:

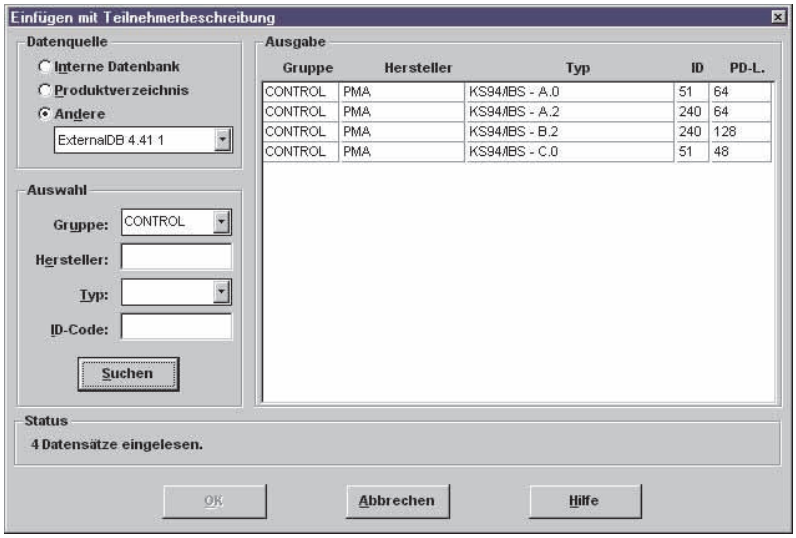
- Read-in via the configuration frame



- Insertion with ident code via the connecting module



- Insertion with description via the connecting module



The two possible methods for introducing the KS94 unit description into CMD are.

- Method 1: Copy file exdevdb.mdb from directory A:<ks94ibs<cmd< on the ES disk into target directory C:<ibscmd<bin . This should be done only, if no other external unit descriptions were entered so far.

- Method 2: Import the required unit descriptions into CMD by calling up in menu
”Extras → Importieren” from disk A:

Manufacturer name	Instrument group	Instrument type	Contents
PMA	KS94	A0	Structure A.0 ID: 51
		A2	Structure A.2 ID:240
		B2	Structure B.2 ID:240
		C0	Structure C.0 ID:240

4.2.3 Introducing KS94 instrument logos in CMD

Copy file pma.icl from directory A:<ks94ibs<cmd< on the ES disk into target directory C:<ibscmd<picture.

4.3 Process data modules

The following process data modules can be configured:

The values to be adjusted and data significations are given in the following tables:

Structure A.0

Transmission of digital and analog values directly via the process mapping (without PCP).

Ident-Code: 51 $_{dec}$; length of process data channel: 64 Bit

Byte	IN (KS94 → INTERBUS)	OUT (INTERBUS → KS94)
1 2	Statusword 1	Control word 1
3 4	Statusword 2	Control word 2
5 6	IN 1	OUT 1
7 8	IN 2	OUT 2

Structure A.1

Transmission of digital and analog values directly via the process mapping (with PCP).

Ident-Code: 240 $_{dec}$;length of process data channel: 64 Bit

Byte	IN (KS94 → INTERBUS)	OUT (INTERBUS → KS94)
1 2	Statusword 1	Control word 1
3 4	Statusword 2	Control word 2
5 6	IN 1	OUT 1
7 8	IN 2	OUT 2

Structure B

Transmission of digital and extended analog values directly via the process mapping (with PCP [1)]).

Ident-Code: 240 $_{dec}$;length of process data channel: 128 Bit

Byte	IN (KS94 → INTERBUS)	OUT (INTERBUS → KS94)
1 2	Statusword 1	Control word 1
3 4	Statusword 2	Control word 2
5 6	IN 1	OUT 1
7 8	IN 2	OUT 2
9 10	IN 3	OUT 3
11 12	IN 4	OUT 4
13 14	IN 5	OUT 5
15 16	IN 6	OUT 6

Structure C

Multiplex transmission of digital and analog values.

Ident-Code: 51 $_{dec}$;length of process data channels: 48 Bit

Byte	IN (KS94 → INTERBUS)	OUT (INTERBUS → KS94)
1 2	Statusword1	Control word1
3	Index - Read	Index - Read
4	Index - Write	Index - Write
5 6	Val - Read	Val - Write

Operating principle (read):

- Enter the index number into Index - Read (OUT)
- After the index number is mirrored in Index - Read (IN), the read value is stored in Val - Read.

Operating principle (write):

- Enter the index number into Index - Write (OUT)
- Enter the value to be written into Val - Write (OUT)
- After the index number is mirrored in Index - Write (IN), the value was transmitted.

To ensure consistent data transmission, Index - Write and Val - Write must be updated before an INTERBUS data cycle. If this is not possible, proceed as follows: write '0' into Index – Write, the value to be transmitted into Val - Write and the index number into Index - Write. With entry '0' in Index - Read / Index – Write, data transmission is not started.

4.3.1 Statuswords

Statusword 1

MSB															LSB
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

Bit-no.	Name	Allocation	Status '0'	Status '1'
D0	Auto/Man	Automatic/manual	Automatic	Manual
D1	Coff	Controller switched off	no	yes
D2	y1	Switching output 1	off	on
D3	y2	Switching output 2	off	on
D4	Lim1	Alarm 1	off	on
D5	Lim2	Alarm 2	off	on
D6	Lim3	Alarm 3	off	on
D7	Lim4	Alarm 4	off	on
D8	Fail1	Input 1 Fail	no	yes
D9	'0'	Always '0'	—-	—-
D10	Merr	Module error [1]	no	yes
D11	NAK	Process data channel write error	no	yes
D12	UPD	Parameter changed	no	yes
D13	On / Off	Online / configuration	Online	Configuration
D14	R/L	Remote / Local	Local	Remote
D15	Dex	Changed data [2]	no	yes

Statusword 2

MSB															LSB
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

Bit-no.	Name	Allocation	Status '0'	Status '1'
D0 - D11	di1 - di12	digital inputs 1 - 12	off	on
D12	Fail3	Input 3 Fail	no	yes
D13	Fail4	Input 4 Fail	no	yes
D14	Fail5	Input 5 Fail	no	yes
D15	Fail6	Input 6 Fail	no	yes

1) Basic unit defective or internal interface switched off
2) See chapter Blocking mechanism with changes

4.3.2 Controlwords

Controlword 1

MSB															LSB
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

Bit-no.	Name	Allocation	Status '0'	Status '1'
D0	Auto/Man	Automatic/Manual	Automatic	Manual
D1	Coff	Controller switch off	no	yes
D2	w / W2	w / W2 switch-over	w	W2
D3	We / Wi	Wext/Wint switch-over	Wext	Wint
D4	w / dW	Set-point offset	off	on
D5	y / Y2	Additional correcting value	off	on
D6	OStart	Selftuning Start	off	on
D7	'0'	always '0'	—-	—-
D8 -D11	do1 - do4	Digital Outputs 1 - 4	off	on
D12	OUT4	Output OUT4	off	on
D13	OUT5	Output OUT5	off	on
D14	LOC	Unit switched over to	Remote	Local[1)]
D15	Dval	Data valid, acknowledgement [2)] positive flank '0' → '1'		

Controlword 2

MSB															LSB
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

Bit-no.	Name	Allocation	Status '0'	Status '1'
D0 - D11	di1 - di12	digital inputs 1 - 12	off	on
D12	OUT1	output OUT1	off	on
D13	OUT2	output OUT2	off	on
D14	do5	digital output 5	off	on
D15	do6	digital output 6	off	on

4.3.3 Locking mechanism with changes

Changing the reference to a datum to be transmitted during operation, e.g. online via parameter channel or via the engineering interface implies a risk that the values are misinterpreted by bus master and KS94. This shall be prevented by means of a locking mechanism.

- When changing a reference, the controller module sets bit Dex = 1.
- The master must evaluate bit Dex
- Acknowledgement and the statement that there are now valid write data on the master side are generated via a positive flank for bit Dval.
- When receiving a positive flank, the controller module sets Dex = 0 and takes over the sent data.
- Resetting Dex is also possible by voltage switch-on and off.

1) Inverse signification as in the status information – Default setting is remote (user need not do anything)
2) See section Blocking mechanism with changes

4.4 Data_types

Internal KS94 data are allocated to data types.

- FP[1)]
  Floating point number
  Range: -9999 ... -0.001, 0, 0.001 ... 9999
- INT
  Positive integer number
  Range: 0 ... 32767
  Range with configuration words: 0000 ... 9999 (→ pag)
  Exception: switch-off value '-32000'
- ST1
  Status, bit-oriented, 1 byte length
  Range: 00H ... 3FH, transmit: 40H...7FH
  Only 6 bits for information transmission can be used, i.e. bit 0...5 (LSB = bit 0).
  Bit 6 is always set to '1'.
- CHAR5/16
  Text string comprising n characters, e.g. defined n=1, n=5, n=16
  permissible characters: 20H...7FH
- ICMP (Integer Compact)
  Bit information as integer transmission, max. 15 bits
  Range: 0...32767; integer transmission is in ASCII format.

	Fixed to '0'	Bit signification														
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Value	-	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

Example:
Bit 13 = 1 and bit 1 = 1, all remaining bits are '0'
internal hex value: 0x2002, as integer value: 8194, transmitted ASCII value: '8194'

4.4.1 Mapping on INTERBUS process data channel

- KS94 data types ST1, ICMP, INT are transmitted as value without changing.
  Example: programmer segment time Tp1

Value in KS94	Value via INTERBUS
500	500

- KS94 data types FP: conversion into Fixpoint data type
  Example: set-point

Value in KS94	Value via INTERBUS
123.4	1234

4.4.2 Mapping on INTERBUS parameter channel

- KS94 data types ST1, ICMP, INT are transmitted as a value without changing.
  Example: programmer segment time Tp1

Value in KS94	Value via parameter channel		converted
	Byte 1	Byte 2	
500	**01_{hex}**	**$F4_{hex}$**	**500**

- KS94 data types FP: transmission in IEEE format
  Example: set-point

Value in KS94	Value via parameter channel (IEEE in Motorola format)				converted
	Byte 1	Byte 2	Byte 3	Byte 4	
123.4	**42_{hex}**	**$F6_{hex}$**	**CC_{hex}**	**CD_{hex}**	**123.4**

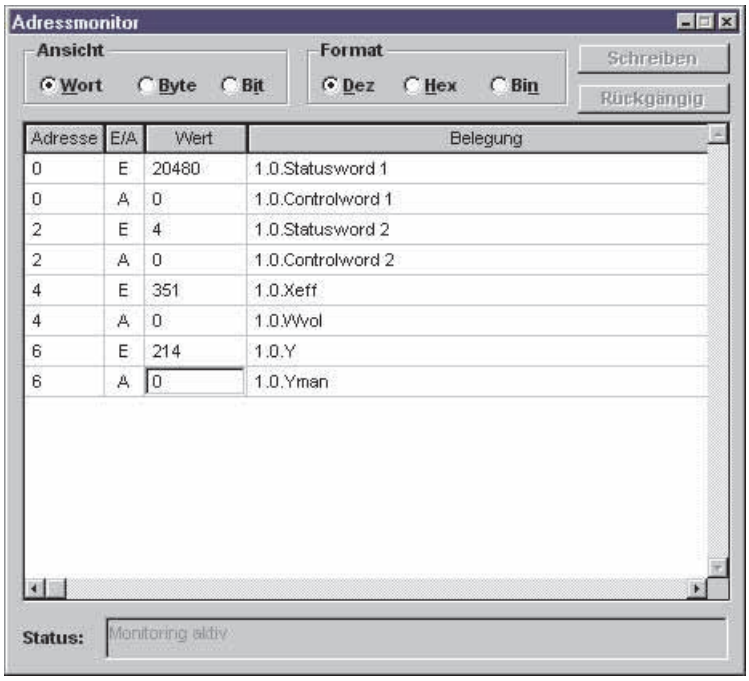
1)

1) Values with variable decimal point (floating point value) are described as FP below.

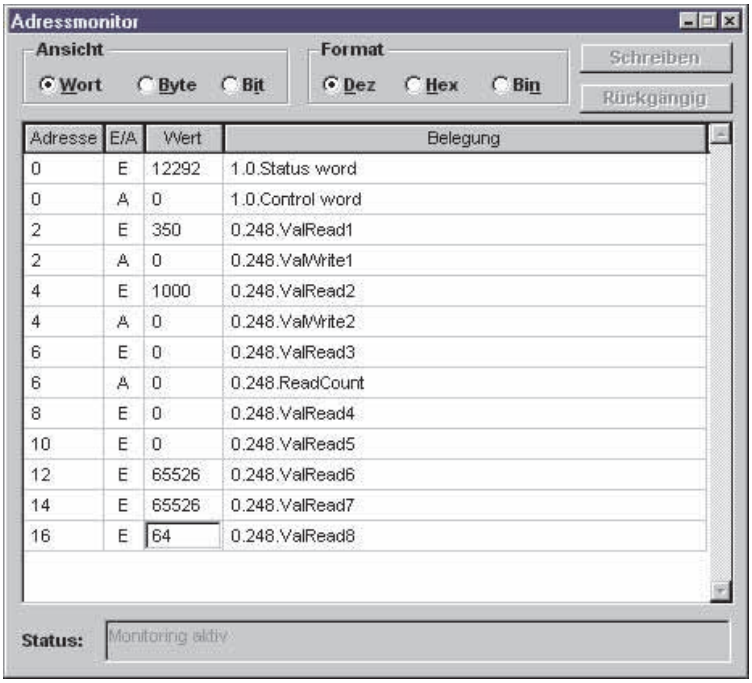
5 Example projects for CMD

Two examples on the engineering set disk are available:

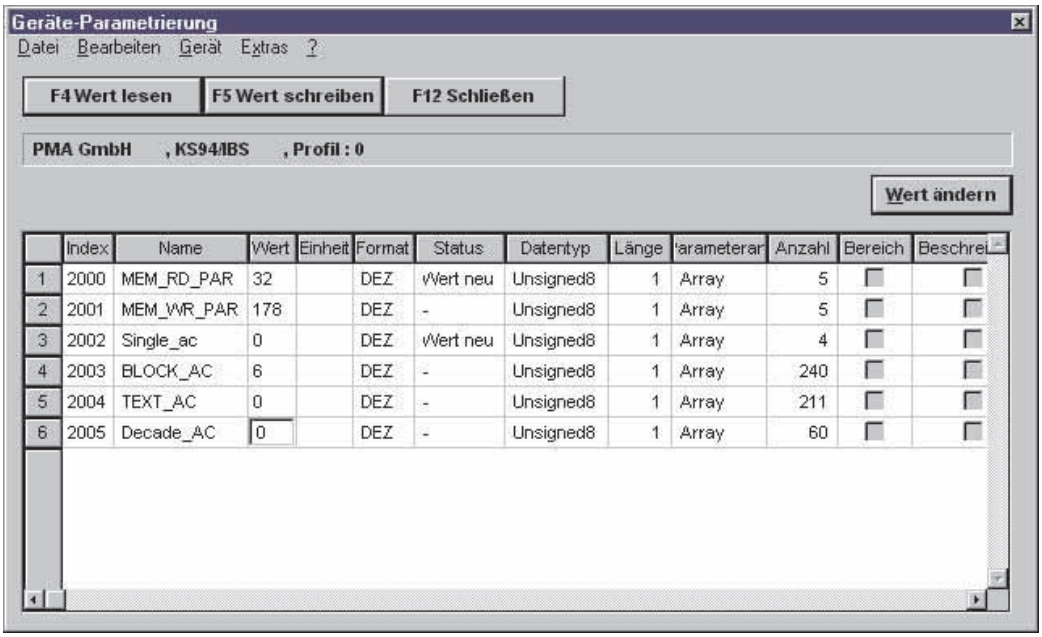
- Demo 1 : standard process data exchange with structure A.0
  transmission of Xeff, Yeff, Wvol, Yman.



- Demo 2 : multiplex example with structure C.0
  During CMD pre-processing, 8 values are read and 2 values are written by the unit and made available to the user as data interface. This project can be matched easily to other numbers of data to be transmitted by the user.



❑ Demo 3 : Standard process data exchange with
structure A.2
Transmission of Xeff, Yeff, Wvol, Yman. + Parameter channel
The address monitor view corresponds to the picture at Demo 1.
For testing the parameter channel, "Instrument parameter setting" is available. The "Instrument parameter setting" can be used for purposeful reading, comparing and writing of the parameter descriptions (PCP objects) of the selected bus unit.



	Index	Name	Wert	Einheit	Format	Status	Datentyp	Länge	Parameterart	Anzahl	Bereich	Beschrei
1	2000	MEM_RD_PAR	32		DEZ	Wert neu	Unsigned8	1	Array	5	☐	☐
2	2001	MEM_WR_PAR	178		DEZ	-	Unsigned8	1	Array	5	☐	☐
3	2002	Single_ac	0		DEZ	Wert neu	Unsigned8	1	Array	4	☐	☐
4	2003	BLOCK_AC	6		DEZ	-	Unsigned8	1	Array	240	☐	☐
5	2004	TEXT_AC	0		DEZ	-	Unsigned8	1	Array	211	☐	☐
6	2005	Decade_AC	0		DEZ	-	Unsigned8	1	Array	60	☐	☐

There are two versions of each project example with different connecting modules :
a) for PC card IBS SC / I-T (e.g. K94dmo1a.bg4)
b) for Siemens S5 IBS S5 DSC/I-T (e.g. K94dmo2b.bg4)

5.1 Example loading procedure

- Make the connection (→ see also section 2.3 page 8).
- Load the KS94 with a suitable engineering (from ET/KS94 version 4.3 SR1). For this, an example from the ES disk can be used.
  File on ES disk: A:\ks94ibs\et94\ibsdemo1.e94 or
  A:\ks94ibs\et94\ibsdemo2.e94
  A:\ks94ibs\et94\ibsdemo3.e94

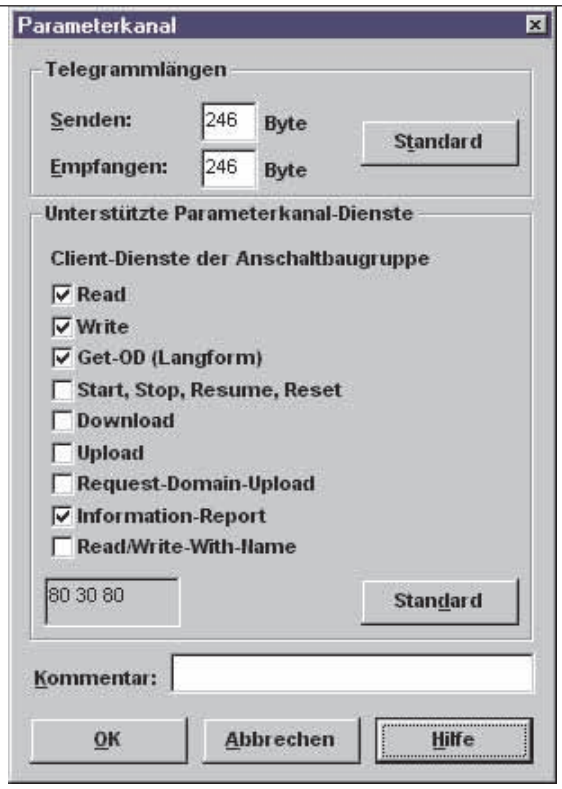
ⓘ The example must be matched to hardware and application. Indispensable settings for the example are structure selection and order of field bus data (→ see section 4.2.1 page 11).

- Switch KS94 to Remote
- Copy the CMD program examples from A:\ks94ibs\cmd\demo1\k94dmo1a.* or \k94dmo1b.* and
  A:\ks94ibs\cmd\demo2\k94dmo2a.* or \k94dmo2b.*
  into target directory c:\ibscmd\project*.*
- Call up CMD and load the required project into the connecting module with execution of pre-processing.
- If necessary, activate the KS94 structure switch-over via menu ”Configuration/Bedienen/Alarm-stop” and ”Configuration/Bedienen/Datenübertragung starten” .
- Go to the ‘Monitoring’ mode. Select the address monitor and handle process data.
- If necessary, set bit D15 in control word and reset it to remove write protection.

6 Parameter transmission (PCP)

For parameter transmission, ‘parameter channel (PCP)’ is available for transparent data exchange via the function block protocol independent of which controller is used. Thereby, all possible access modes of the protocol are supported (single access, block of ten and overall block). Communication to the controller is transparent, i.e. the user himself is responsible for monitoring the value ranges, operating modes (remote/local) etc. The PCP channel uses 2 words in the INTERBUS ring. KS94 supports PCP version 2.0.

Fig.: 6 KS94-IBS parameter channel



6.1 Supported PCP services:

❑ Context management

- Initiate — Build up communication
- Abort — Cancel communication
- Reject — Service cancelation with errors

❑ VFD-Support (Virtual Field Device):

- Status — Read instrument status
- Identify — Read instrument type or instrument version and firmware version

PMA GmbH
KS94/IBS
SW x.x HW x.x

❑ Variable Access:

- Read — Read access to objects
- Write — Write access to objects

ⓘ Further information for realization of the communication services in your application is given in the user manual of your connecting module.

6.2.2 Building up a communication

❑ **Initiate**

Service "Initiate-Request" is used for building up a communication between two communication partners. When building up the communication, the settings of both bus units for the sizes of send and receive buffer and for the supported services are checked for compliance.

word		Request			Confirmation			Confirmation (on case of error)	
1	*Initiate*	00	8B	*Initiate*	80	8B		80	8B
2		00	02		00	06		00	08
3		00	**CR**		00	**CR**		00	**CR**
4		00	00		**Result (+)**			**ErrClass**	**ErrCode**
5					00	00		**AddCode**	
6					00	00		00	40
7					FF	00		00	40
8					00	00		?	?
9								?	?
10								?	?

6.2.3 Sending data to the bus unit

❑ **Write**

Use service "Write-Request" for sending a data field from your connecting module to KS94-IBS.

Word		Request			Confirmation			Confirmation (in case of error)	
1	*Write*	00	82	*Write*	80	82		80	82
2		**Numb.following DW**			00	02		00	02
3		00	**CR**		00	**CR**		00	**CR**
4		**Index**			**Result (+)**			**ErrClass**	**ErrCode**
5		**Subindex**	**Length**					**AddCode**	
6		**Data ...**							
7									
8									

6.2.4 Reading data from a bus unit

❑ **Read**

Use service "Read-Request" for reading data from KS94-IBS into a data field on your connecting module.

Word		Request			Confirmation			Confirmation (in case of error)	
1	*Read*	00	81	*Read*	80	81		80	81
2		00	03		00	xx		00	03
3		00	**CR**		00	CR		00	CR
4		**Index**			**Result (+)**			**ErrClass**	**ErrCode**
5		**Subindex**	-		-	**Length**		**AddCode**	
6					**Data ...**				
7									
8									

6.2.5 Terminating the communication

❑ **Abort**

Use service "Abort-Request" to terminate the communication between your connecting module and KS94-IBS. **Service *'Abort'* is not confirmed !**

Word		Request	
1	*Abort*	00	81
2		00	03
3		-	**C R**
4		00	00
5		00	00

6.3 KS94-IBS objects

6.3.1 Survey

The following table contains all KS94-IBS objects which are required for a data exchange via the parameter channel.

Index	Object name	Object description	Access
2000_{hex}	MEM_RD_PAR	Object 'MEM_RD_PAR' is used for storage of transmission parameters for all read accesses to KS94 via the parameter channel.	R/W
2001_{hex}	MEM_WR_PAR	Object 'MEM_WR_PAR' is used for storage of transmission parameters for all write accesses to KS94 via the parameter channel.	R/W
2002_{hex}	SINGLE_AC	Object 'SINGLE_AC' contains the data for an individual access.	R/W
2003_{hex}	BLOCK_AC	Object 'BLOCK_AC' contains the data for a block access.	R/W
2004_{hex}	TEXT_AC	Object 'TEXT_AC' contains the data for a text access.	R/W
2005_{hex}	DECADE_AC	Object 'DECADE_AC' contains the data for a tens block access.	R/W

For data exchange with KS94-IBS via the parameter channel, the following procedure is required:

❑ Reading values:

- Enter the parameters of the values to be transmitted with a 'Write Request' into object 2000_{hex} (MEM_RD_PAR) (→ see examples on the following pages).
- Read data with a 'Read Request' to the relevant object (SINGLE_AC; BLOCK_AC; TEXT_AC or DECADE_AC).

❑ Writing value:

- Enter the parameters of the values to be transmitted with a 'Write Request' into object 2001_{hex} (MEM_WR_PAR) (→ see examples on the following pages).
- Write data with a 'Write Request' to the relevant object (SINGLE_AC; BLOCK_AC; TEXT_AC or DECADE_AC).

6.3.2 Transmission parameters

The transmission parameters are given in the function block protocol (→ section page ff). The transmission parameters are composed of:

- CODE_RD/WR ≙ 'Code' in the FB protocol
  The code identification is within a range of $00...99_{dec}$ and $B2_{hex}(178_{dec})$ and $B3_{hex}(179_{dec})$
- FB_NO_RD/WR ≙ FB no. in the FB protocol
  A function block is addressed with a function block number which is within range ‘0’ and ‘250’ .Function block number ranges:
  - 0 general data for the overall instrument
  - 1 - 99 fixed function blocks
- FCT_NO_RD/WR ≙ Fct-no. in the FB protocol
  A function as a partial address of a function block is addressed by means of a function number, which is within range ‘0’ and ‘99’.
- TYPE_NO_/WR ≙ type no. in the FB protocol
- DATA_FORMAT_RD/WR (used only with SINGLE_AC; integer = 0, real = 1)

6.3.3 Object 2000_{hex} 'MEM_RD_PAR'

Object 'MEM_RD_PAR' is used for storage of transmission parameters for all read accesses to KS94 via the parameter channel.

Object name	**MEM_RD_PAR**
Object code	Array
Access	WRITE_ALL / READ_ALL
Data type index	USIGNED8 (5)
Parameter	
• Index	2000_{hex}
• Subindex	Subindex 0 can be used for addressing all 5 elements of the ARRAY, subindex 1…5 can be used for addressing individual elements of the ARRAY.
• Length	5 (bytes)
• Data	1: CODE_RD 2: FB_NO_RD 3: FCT_NO_RD 4: TYPE_NO_RD 5: DATA_FORMAT_RD (only used with object 'SINGLE_AC'; integer = 0, real = 1)

6.3.4 Object 2001_{hex} 'MEM_WR_PAR'

Object 'MEM_WR_PAR' is used for storage of transmission parameters for all write accesses to KS94 via the parameter channel.

Object name	**MEM_WR_PAR**
Object code	Array
Access	WRITE_ALL / READ_ALL
Data type index	USIGNED8 (5)
Parameter	
• Index	2001_{hex}
• Subindex	Subindex 0 can be used for addressing all 5 elements of the ARRAY, subindex 1…5 can be used for addressing individual elements of the ARRAY.
• Length	5 (bytes)
• Data	1: CODE_RD 2: FB_NO_RD 3: FCT_NO_RD 4: TYPE_NO_RD 5: DATA_FORMAT_RD (used only with SINGLE_AC; integer = 0, real = 1)

6.3.5 *Object 2002_{hex} 'SINGLE_AC'*

Object 'SINGLE_AC' contains the data for an individual access.

Object name	**SINGLE_AC**			
Object code	STRING_VAR_OBJECT			
Access	WRITE_ALL / READ_ALL			
Data type index	OCTET_STRING ($0A_{hex}$)			
Parameter				
• Index	2002_{hex}			
• Subindex	0			
• Length	4 (Byte)			
• Data	Value to be transmitted, with selection real (data format 01,xx)			
	Byte 0: Real value, hwB [1)]	Byte 1: Real value, hwB-1	Byte 2: Real value, hwB-2	Byte 3: Real value, hwB-3
	Value to be transmitted, with selection integer (data format 00,xx)			
	Byte 0: Integer value, hwB		Byte 1: Integer value, hwB-1	

Example: The internal set-point (Wvol) shall be read and written as a real value.

Service 'Write' is used for assigning the transmission parameters for datum 'Wvol' (code = 32; FB no. = 50; Fct no. = 1 and type no. = 90) to objects MEM RD PAR and MEM WR PAR see page .

Request

No.	Value	Format	Comment
1	0, 2	dec, dec	Invoke_ID, CR
2	2000/2001	hex	Index (MEM_RD_PAR / MEM_WR_PAR)
3	0,5	dec, dec	Subindex, Data_length (5Byte)
4	32, 50	dec, dec	Code, FB no.
5	1, 90	dec, dec	Fct no., type no.
6	1,0	dec, dec	Data format (0 = integer; 1 = real), ---

Confirmation

No.	Value	Format	Comment
1	0,2	dec, dec	Invoke_ID, CR
2	0000	hex	Parameter

Service 'Write' can be used for writing the Wvol.

Request

No.	Value	Format	Comment
1	0, 2	dec, dec	Invoke_ID, CR
2	2002	hex	Index (SINGLE_AC)
3	0,4	dec, dec	Subindex, Data_length (4 bytes)
4	425c	hex	Wvol
5	0000	hex	

Confirmation

No.	Value	Format	Comment
1	0,2	dec, dec	Invoke_ID, CR
2	0000	hex	Parameter

Service 'Read' can be used for reading the Wvol.

Request

No.	Value	Format	Comment
1	0, 2	dec, dec	Invoke_ID, CR
2	2002	hex	Index (SINGLE_AC)
3	0,0	dec, dec	Subindex,---

Confirmation

No.	Value	Format	Comment
1	0,2	dec, dec	Invoke_ID, CR
2	0000	hex	Parameter
3	0004	hex	Data_length (4 bytes)
4	425c	hex	Wvol
5	0000	hex	

1) hwB (most significant byte)

6.3.6 *Object 2003_{hex} 'BLOCK_AC'*

Object 'BLOCK_AC' contains the data for a block access.

Object name	**BLOCK_AC**			
Object code	STRING_VAR_OBJECT			
Access	WRITE_ALL / READ_ALL			
Data type	OCTET_STRING ($0A_{hex}$)			
Parameter				
• Index	2003_{hex}			
• Subindex	0			
• Length	max. 240 (bytes)			
• Data	Byte 0: number of REAL values		Byte 1: number of integer values	
	Byte 2: 1st real value, hwB [1)]	Byte 3: 1st real value, hwB-1	Byte 4: 1st real value, hwB-2	Byte 5: 1st real value, hwB-3
	Byte 6: 2nd real value, hwB	Byte 7: 2nd real value, hwB-1	Byte 8: 2nd real value, hwB-2	Byte 9: 2nd real value, hwB-3
	...			
	Byte x: 1st real value, hwB	Byte x+1: 1st real value, hwB-1	Byte x+2: 1st real value, hwB-2	Byte x+3: 1st real value, hwB-3
	Byte x+4: 1st integer value, hwB		Byte x+5: 1st integer value, hwB-1	
	Byte x+6: 2nd integer value, hwB		Byte x+7: 2nd integer value, hwB-1	
	...			
	Byte z: uth integer value, hwB		Byte z+1: uth integer value, hwB-1	

Example: The correcting variable parameters (Ymin; Ymax; Y2 and Y0) must be read.

Service 'Write' is used for assigning the transmission parameters for the correcting variable parameters to object MEM_RD_PAR (code = 178[B2]; FB no. = 50; Fct. no. = 4 and type no. = 90) see page .

Request

No.	Value	Format	Comment
1	0, 2	dec, dec	Invoke_ID, CR
2	2000	hex	Index (MEM_RD_PAR)
3	0,5	dec, dec	Subindex, Data_length (5 bytes)
4	178, 50	dec, dec	Code, FB no.
5	4, 90	dec, dec	Fct no., type no.
6	0,0	dec, dec	Data format, ---

Confirmation

No.	Value	Format	Comment
1	0,2	dec, dec	Invoke_ID, CR
2	0000	hex	Parameter

Service 'Read' is used for value reading.

Request

No.	Value	Format	Comment
1	0, 2	dec, dec	Invoke_ID, CR
2	2003	hex	Index (BLOCK_AC)
3	0,0	dec, dec	Subindex,---

Confirmation

No.	Value	Format	Comment
1	0,2	dec, dec	Invoke_ID, CR
2	0000	hex	Parameter
3	0012	hex	Length 18 bytes
4	0400	hex	4 real values; 0 integer values
5	0000	hex	Ymin
6	0000	hex	
7	42c8	hex	Ymax
8	0000	hex	
9	0000	hex	Y2
10	0000	hex	
11	4248	hex	Y0
12	0000	hex	

1) hwB (most significant byte)

6.3.7 *Object 2004_{hex} 'TEXT_AC'*

Object 'TEXT_AC' contains the data for a text access.

Object name	**TEXT_AC**	
Object code	STRING_VAR_OBJECT	
Access	WRITE_ALL / READ_ALL	
Data type index	OCTET_STRING ($0A_{hex}$)	
Parameter		
• Index	2004_{hex}	
• Subindex	0	
• Length	max. 240 (bytes)	
• Data	Byte 0 = 0	Byte 1: NUMBER INTEGER
	Byte 2...byte 211: variable length	

Example: The display texts shall be written (String1 = 'Heating ; Unit = 'Meters').

Service 'Write' is used for assigning the transmission parameters for the display texts (code = 178[B2]; FB no. = 0; Fct no. = 1 and type no. = 0) to object MEM_WR_PAR see page .

Request

No.	Value	Format	Comment
1	0, 2	dec, dec	Invoke_ID, CR
2	2001	hex	Index (MEM_WR_PAR)
3	0,5	dec, dec	Subindex, Data_length (5 bytes)
4	178, 0	dec, dec	Code, FB no.
5	1, 0	dec, dec	Fct no., type no.
6	0,0	dec, dec	Data format, ---

Confirmation

No.	Value	Format	Comment
1	0,2	dec, dec	Invoke_ID, CR
2	0000	hex	Parameter

Service 'Write' is used for writing the display texts. String 2 to String12 are filled with spaces (20_{hex}). The individual strings are separated by comma ($2C_{hex}$).

Request

No.	Value	Format	Comment
1	0, 2	dec, dec	Invoke_ID, CR
2	2004	hex	Index (TEXT_AC)
3	0,211	dec, dec	Subindex, Data_length (211 bytes)
4	0,13	dec, dec	0 real, 13 texts
5	4865	hex	Data (H e)
6	697a	hex	Data (i z)
7	656e	hex	Data (e n)
8	2020	hex	Data (Space Space)
9	2020	hex	Data (Space Space)
10	2020	hex	Data (Space Space)
11	2020	hex	Data (Space Space)
12	2020	hex	Data (Space Space)
13	2c..	hex	
…			
106	..2c	hex	
107	4d65	hex	Data (M e)
108	7465	hex	Data (t e)
109	7200	hex	Data (r)

Confirmation

No.	Value	Format	Comment
1	0,2	dec, dec	Invoke_ID, CR
2	0000	hex	Parameter

6.3.8 Object 2005_{hex} 'DECADE_AC'

Object 'DECADE_AC' contains the data for a tens block access. All data are sent in real format.

Object name	**DECADE_AC**			
Object code	STRING_VAR_OBJECT			
Access	READ_ALL			
Data type index	OCTET_STRING ($0A_{hex}$)			
Parameter				
• Index	2005_{hex}			
• Subindex	0			
• Length	max. 60			
• Data	Byte 0: number of REAL values (max. 9)		Byte 1: number of integer values (fixed to 0)	
	Byte 2 1st real value, hwB [1]	Byte 3 1st real value, hwB-1	Byte 4 1st real value, hwB-2	Byte 5 1st real value, hwB-3
	Byte 6 2nd real value, hwB	Byte 7 2nd real value, hwB-1	Byte 8 2nd real value, hwB-2	Byte 9 2nd real value, hwB-3
	...			
	Byte x yth real value,hwB	Byte x+1 yth real value, hwB-1	Byte x+2 yth real value, hwB-2	Byte x+3 yth real value, hwB-3

Example: Set-points (Wnvol, Wvol und Wdw) shall be written and read.

Service 'Write' is used for assigning the transmission parameters for the set-points (code = 30; FB no. = 50; Fct no. = 1 and type no. = 90) to object MEM_RD_PAR see page .

Request

No.	Value	Format	Comment
1	0, 2	dec, dec	Invoke_ID, CR
2	2000	hex	Index (MEM_RD_PAR)
3	0,5	dec, dec	Subindex, Data_length (5 bytes)
4	30,50	dec, dec	Code, FB no.
5	1, 90	dec, dec	Fct no., type no.
6	0,0	dec, dec	Data format, ---

Confirmation

No.	Value	Format	Comment
1	0,2	dec, dec	Invoke_ID, CR
2	0000	hex	Parameter

Service 'Read' can be used for reading the set-points.

Request

No.	Value	Format	Comment
1	0, 2	dec, dec	Invoke_ID, CR
2	2005	hex	Index (DECADE_AC)
3	0,0	dec, dec	Subindex,---

Confirmation

No.	Value	Format	Comment
1	0,2	dec, dec	Invoke_ID, CR
2	0000	hex	Parameter
3	000e	hex	Data_length (4 bytes)
4	0300	hex	Number of real values (3) integer values (0)
5	4495	hex	Wnvol
6	8000	hex	
7	4495	hex	Wvol
8	8000	hex	
9	0000	hex	WdW
10	0000	hex	

1) hwB (most significant byte)

6.4 PCP error messages

Unless an action could be realized successfully by means of a service, e.g. Initiate, Read or Write, an error message in parameters Error Class, Error Code and Additional Code is provided in addition to the negative confirmation.

A detailed list of all PCP error messages is given in the user manual of your connecting module.

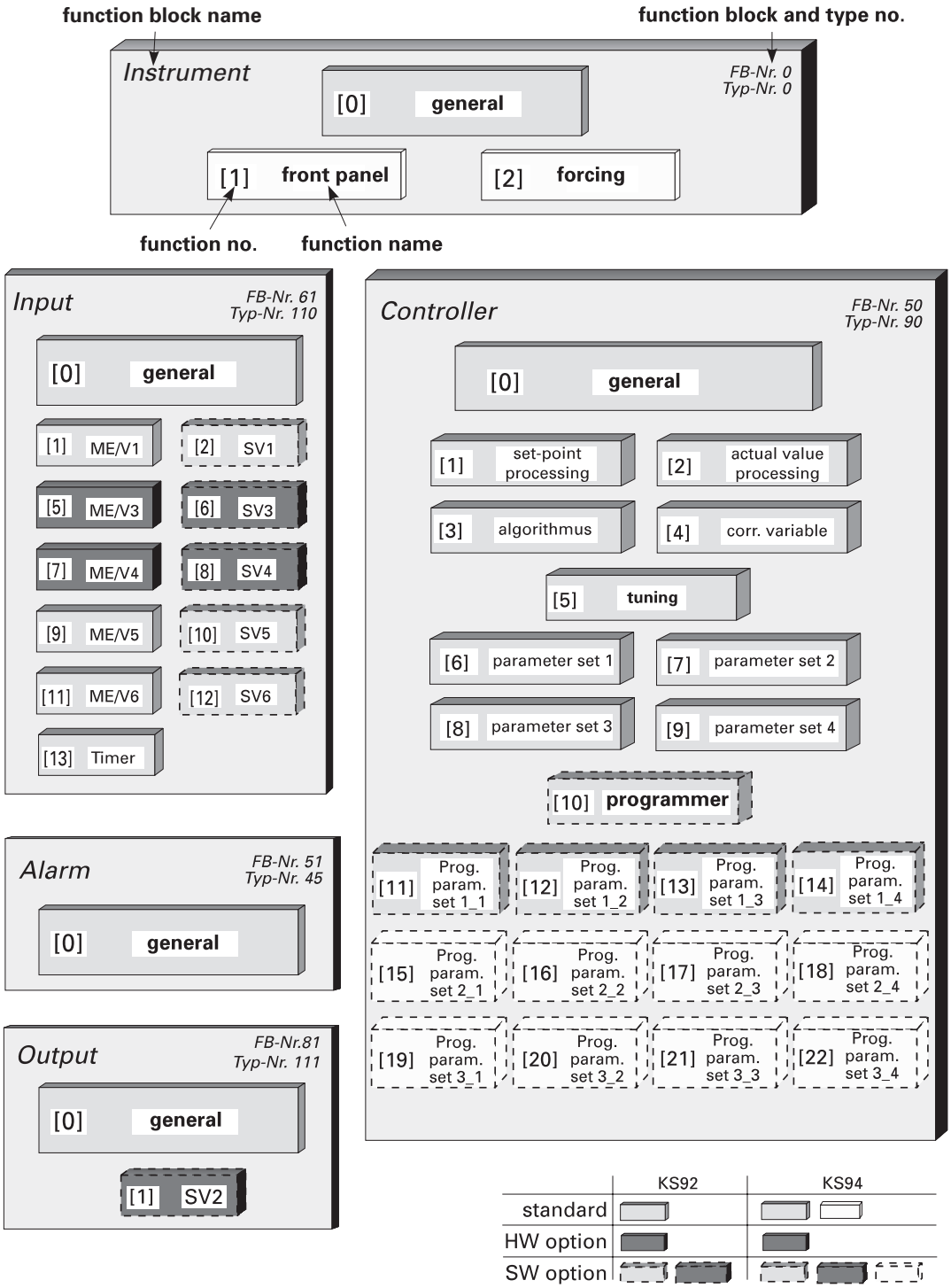
Error Class	Error Code	Add. Code	Signification	Cause
8	0	01_{hex}	Timeout	KS94 is set to front-panel interface, or is defective.
8	0	02_{hex}	Parity error	Internal communication error
8	0	03_{hex}	BCC error	Internal communication error
8	0	04_{hex}	NAK	KS94 does not accept the transmitted data (data out-of-range ...)
8	0	05_{hex}	Logic error in the transmission parameters	

7 Function block protocol

7.1 Data structuring

Due to the multitude of information to be processed in KS94, logically related data and actions are classified into function blocks. Five function blocks are defined for KS94. They are addressed via fixed block addresses. Each block is divided into individual functions which are provided dependent of HW or SW options. Functions are numbered per function block. Function number 0 addresses function block-specific data.

Fig.: 7 Survey of function blocks and functions of KS94



Principles of the function block protocol
A function block comprises input and output data (process data) as well as parameter and configuration data. It can be addressed via a block number. An allocated block type defines the relevant function.
The various access mechanisms are:

Single access

- This access (code xx) can be used for reading or writing a single process value of a function. Single accesses to parameter and configuration data are not possible.

Block access (tens block)

- This access (code x0) can be used for reading max. nine process values of a function.

Block access (overall block)

- This access can be used for reading and writing all parameter (code B2) and configuration data (code B3) of a function. The following conditions are applicable for this access: For writing data with 'Code B3', the unit must be switched to configuration mode (Ä see page 'OpMod'). All new configuration data and parameters entered are only effective, when the unit was switched back to on-line. All data of a message must be defined, omissions are not permissible. If all parts of a message in the instrument are unused (HW and SW options), the complete message must be transmitted nevertheless. Checking the non-available data is omitted. With faulty block write accesses, the following rule is applicable: A message is answered with NAK, if min. one datum is faulty. Already valid values are stored.

7.2 CODE tables

7.2.1 Configuration words (C.xxxx)

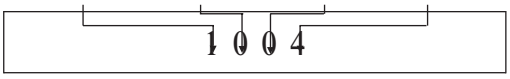
The configuration words given in the following code tables comprise several partial components, which can be transmitted only in common. The data in the table must be interpreted as follows:

Example (C100):

Code	Descr.	R/W	Type	Description	Range
B3	C100	R/W	INT	CFunc:controller function (T,H) CType:controller type (Z) WFunc:set-point function (E)	0..xxyz

Description	CFunc		CType	WFunc
	Thousands	Hundreds	Tens	Ones
Range	x	x	y	z
	00 ... 12		0...4	0...7

Example: continuous controller; standard controller; Set-point/cascade with offset



For transmission of configuration words, see section page 17.

7.2.2 INSTRUMENT (FB no.: 0 type no.: 0)

All data valid for the overall instrument are classified in function block 'INSTRUMENT'.

Processdata

General (Function-no: 0)

Code	Descr.	R/W	Type	Description	Range	Bem.
01	Unit_State 1	R	ST1	Status 1		**A**
10	Block 13..15, 18	R	Block			
13	Write Error	R	INT	Error during last write access	0, 100...127	
14	Write Error Position	R	INT	Position of last write access error	0...99	
15	Read Error	R	INT	Error during last read access	0, 100...127	
18	Type	R	INT	Type no. of function block	0	
20	Block 21...27	R	Block			
21	HWbas	R	INT	Basic HW options: module A, P		**B**
22	HWext	R	INT	Ext. HW options: module B, C		**C**
23	SWopt	R	INT	SW options 1		**D**
24	SWcod	R	INT	SW code no. 7th to 10th digit of 12NC	wxyz	**E**
25	SWvers	R	INT	SW code no. 11th to 12th digit of 12NC	00xy	**F**
26	OPVers	R	INT	Operating version		
27	EEPVers	R	INT	EEPROM version		
31	OpMod	R/W	INT	Switch over controller to configuration mode (only after 1)	0	
				Switch over controller to on-line mode (only after 0)	1	
				Cancelation of configuration mode (only after 0)	2	
32	Local-Switch	R/W	INT	Switch-over to local mode	0..1	**G**
33	UPD	R/W	INT	Local data change acknowledgement	0..1	**H**
39	BSAct	(R)/ W	INT	Activation of process data structure IBS	0..1	**I**

Rem A ***Unit_State1***

MSB							LSB
D7	D6	**D5**	D4	D3	**D2**	D1	**D0**

Bit no.	Name	Allocation	Status '0'	Status '1'
D0	R/W	Controller status	Local	Remote
D1	CNF	Controller status	online	configuration
D2...D4	'0'	Always '0'		
D5	UPD	Parameter Update	No	Yes
D6	'1'	Always '1'		
D7		Parity		

Rem B ***HWbas***

	Instrument type (module A)		output- HW (module P)		
	T	H	Z	E	
KS92	01		01		Relay: Out1,2,4,5
KS94	11		11		Current: Out1, Relay: Out2,4,5

Example: Value 'HWbas = 1111' means that the addressed unit is a KS94 with 3 relays and 1 current output. (12NC e.g. 9407 924xx xxx oder 9407 928xx xxx).

Rem C ***HWext***

Module B		Module C		
T	H	Z	E	
Not provided	00*	00*		Not provided
TTL-Interface	01	01		A) 1 analog output (continous) (OUT3)
RS485/422-Interface	02	02		B) 2 analog inputs (INP3, INP4)
PROFIBUS	10	04		C)5 digital inputs (DI8...DI12 and 2 digital outputs (DO5...DO6)
INTERBUS	11			
		05		A + C
		06		B + C
		07		A + B + C

* Default setting

Example: Value ‘HWext = 104‘ means the addressed instrument is equipped with a module B as TTL interface without real-time clock and a module C with version c . (12NC e.g. 9407-9xx-16xxx).

Rem D ***SWopt Conversion 12NC 10th digit***

T				H				Z				E			
0	EXT	0	0	0	0	0	0	0	0	0	SOPT	0	PRG	MWK	SV

Descr.	Status '0'	Status '1'
SV	Signal processing blocked	Signal pocessing enabled
MWK	Measurement value correction f. temperature measurement disabled	Measured value correction f. temperatur e measurement enabled
PRG	Programmer disabled	Programmer enabled
SOPT	Optimization at set-point disabled	Optimization at set-point enabled
EXT	Standard	Extension enabled

Example: Value ‘SWopt = 13‘ means that options signal processing, measured value correction and optimization at the set-point are enabled. (12NC e.g. 9407 9xxxx 3xx).

Rem E ***SWCod***

T	H	Z	E
7th digit	8th digit	9th digit	10th digit

Example: Value ‘SWCod= 7239‘ means that the addressed instrument contains software code number 4012 157 239xx.

Rem F ***SWvers***

T	H	Z	E
0	0	11th digit	12th digit

Example: Value ‘SWVers= 11‘ means that the addressed instrument contains software code number 4012 15x xxx11.

Rem G ***Local-Switch***

Switch-over to Local mode (only valid with digital input di3 REMOTE closed):

0: switch off local (default)

1: switch on local

Rem H ***UPD***

UPD has value 1, if parameters or configuration data were changed by local operation, or after power-on.

Rem I ***BSAct***

Activate INTERBUS process data structure switch-over.
The datum is reset to 0 automatically (internal trigger function).

Writing the command can lead to standstill of the INTERBUS and affect the safety of the overall system in an accidental manner.

Forcing (Funktions-Nr: 2)

Code	Descr..	R/W	Type	Description	Range	Rem.
30	Block 31...39	R	Block			
31	FInp 1	R/W	FP	Forced Inp 1 (signal input before measured value correction for INP1)		
32	FInp 3	R/W	FP	Forced Inp 3 (signal input before signal pre-processing)		
33	FInp 4	R/W	FP	Forced Inp 4 (signal input before signal pre-processing)		
34	FInp 5	R/W	FP	Forced Inp 5 (signal input before signal pre-processing)		
35	FInp 6	R/W	FP	Forced Inp 6 (signal input before signal pre-processing)		
36	Fdi	R/W	ICMP	Forced digital inputs di1...di12		**J**
37	FOut 1	R/W	FP	Forced Out 1		**K**
38	FOut 3	R/W	FP	Forced Out 3 (signal input before post-processing)		
39	Fdo	R/W	ICMP	Forced digital outputs Out 1...Out5; do1...do6 (also used for output disabling with corresponding configuration. 0 = enabled; 1 = disabled)		**L**

Rem J ***Data structure***

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Signification	0	0	0	0	di12	di11	di10	di9	di8	di7	di6	di5	di4	d.c.	di2	di1

Rem K ***Range***

-999 … 9999 (0/4 .. 20mA according to the output scaling)
With OUT_X configured as digital output (relay or logic), pre-defined analog values are ineffective, OUT_X can be reached only via Fdo.

Rem L ***Data structure***

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Signification	0	0	0	0	0	do6	do5	do4	do3	do2	do1	Out5	Out4	Out3	Out2	Out1

ComRead (Function no.: 4)

Code	Descr.	R/W	Type	Description	Range	Default	Rem.
20	Block 21...29	R	Block	Block access	21...29		
21	Val 1	R	BCD	Value 1			
22	Val 2	R	BCD	Value 2			
23	Val 3	R	BCD	Value 3			
24	Val 4	R	BCD	Value 4			
25	Val 5	R	BCD	Value 5			
26	Val 6	R	BCD	Value 6			
27	Val 7	R	BCD	Value 7			
28	Val 8	R	BCD	Value 8			
29	Val 9	R	BCD	Value 9			
30	Block 31...37, 39	R	Block	Block access	31...37, 39		
31	Val 10	R	BCD	Value 10			
32	Val 11	R	BCD	Value 11			
33	Val 12	R	BCD	Value 12			
34	Val 13	R	BCD	Value 13			
35	Val 14	R	BCD	Value 14			
36	Val 15	R	BCD	Value 15			
37	Val 16	R	BCD	Value 16			
39	ResetRead	R/W	INT	Configuration allocation reset	0: d.c. 1: Reset[1)]	0	

1) Value will be reset to 0 after handling

ComWrite (Function no.: 5)

Code	Descr.	R/W	Type	Description	Range	Default	Rem.
20	Block 21...29	R	Block	Block access	21...29		
21	Val 1	(R)/W	BCD	Value 1			
22	Val 2	(R)/W	BCD	Value 2			
23	Val 3	(R)/W	BCD	Value 3			
24	Val 4	(R)/W	BCD	Value 4			
25	Val 5	(R)/W	BCD	Value 5			
26	Val 6	(R)/W	BCD	Value 6			
27	Val 7	(R)/W	BCD	Value 7			
28	Val 8	(R)/W	BCD	Value 8			
29	Val 9	(R)/W	BCD	Value 9			
30	Block 31...39	R	Block	Block access	31...39		
31	Val 10	(R)/ W	BCD	Value 10			
32	Val 11	(R)/W	BCD	Value 11			
33	Val 12	(R)/W	BCD	Value 12			
34	Val 13	(R)/W	BCD	Value 13			
35	Val 14	(R)/W	BCD	Value 14			
36	Val 15	(R)/W	BCD	Value 15			
37	Val 16	(R)/W	BCD	Value 16			
38	BusConf	R/W	INT	Selection of the bus process data module	0...3	1	**M**
39	ResetRead	R/W	INT	Configuration allocation reset	0: d.c. 1: Reset [1)]	0	

Rem M ***IBS structure allocations***

0: structure A.0 ”standard function”
1: structure A.1 ”standard function with PCP 2 words” (default)”
2: structure B ”extended function with PCP 2 word””
3: structure C ”flexible selection function”

A change of this structure is activated only after switching on the power supply again, or setting the activation (s. below). [2)]

1) Value will be reset to 0 after handling
2) A change of structure is followed by a data length change on the INTERBUS. This means short-time interruption of the overall bus. Therefore, this may be done by the user only provided that all consequences were taken into account.

Parameter- a. configuration data

General

(Function no.: 0)

Code		Descr.	R/W	Type	Description		Range	Rem.
B2		F_{Key}	R/W	INT	Function of front-panel key		0 .. 2	
		Lock	R/W	INT	EBloc: Block changing extended operating level HBloc: Block auto/manual key CBloc: Block controller switch-off WBloc: Block set-point adjustment	(T) (H) (Z) (E)	0 .. wxyz	
		Disp2	R/W	INT	PBloc: Block program preset RBloc: Block programmer Programmer run/stop/reset OBloc: Block self-tuning	(T) (H) (Z)	0 .. xyz0	
B3		C900 [1]	R/W	INT	Prot: Protocol type Baud: Baudrate (d.c. With PROFIBUS and INTERBUS	(T) (H)(Z)	0...xyy0	
		Addr[1]	R/W	INT	Instrument adress: ISO1745 PROFIBUS		0...99 1...127	
		C902	R/W	INT	Freq: Mains frequency 50/60	(T)	0...x0yz	
		C800	R/W	INT	Text2: Signification of display field text2 UsrTx: User text selection LED: Function of front panel LEDs Langu: Language selection of plain text display	(T) (H) (Z) (E)	0...wxyz	
		C801	R/W	INT	Lunit: Unit selection XDisp: Process value display	(T) (H) (Z)	0...xxy0	

Display Texts

User-definable display texts only for KS94 (function no: 1)

Code		Descr.	R/W	Type	Description	Range	Rem.
B2		String1	R/W	CHAR16	Display text 1	0x20...0x7F	[2]
		String2	R/W	CHAR16	Display text 2	0x20...0x7F	
		String3	R/W	CHAR16	Display text 3	0x20...0x7F	
		String4	R/W	CHAR16	Display text 4	0x20...0x7F	
		String5	R/W	CHAR16	Display text 5	0x20...0x7F	
		String6	R/W	CHAR16	Display text 6	0x20...0x7F	
		String7	R/W	CHAR16	Display text 7	0x20...0x7F	
		String8	R/W	CHAR16	Display text 8	0x20...0x7F	
		String9	R/W	CHAR16	Display text 9	0x20...0x7F	
		String10	R/W	CHAR16	Display text 10	0x20...0x7F	
		String11	R/W	CHAR16	Display text 11	0x20...0x7F	
		String12	R/W	CHAR16	Display text 12	0x20...0x7F	
		Unit	R/W	CHAR5	Unit to specification	20h...7Fh	

String 1 to string 12 must always contain 16 characters and unit must always contain 5 characters (fixed data length!).

1) Baudrate a. address setting are effective only after initialization, e.g. protocol switch-over.
2) For transmitting the user texts via INTERBUS, a data area of min. 211 bytes must be provided.

Forcing					Input and output forcing		(function no.: 2)	
Code		**Descr.**	**R/W**	**Type**	**Description**		Range	Rem.
B3		C910	R/W	INT	Forcing INP1 Forcing INP3 Forcing INP4	(T) (Z) (E)	0..x0yz	
		C911	R/W	INT	Forcing INP5 Forcing INP6	(T) (H)	0...xy00	
		C920	R/S	INT	Forcing di1 Forcing di2 Forcing di4	(T) (H) (E)	0...wx0z	
		C921	R/S	INT	Forcing di5 Forcing di6 Forcing di7 Forcing di8	(T) (H) (Z) (E)	0...wxyz	
		C922	R/S	INT	Forcing di9 Forcing di10 Forcing di11 Forcing di12	(T) (H) (Z) (E)	0...wxyz	
		C930	R/S	INT	Forcing OUT1 Forcing OUT2 Forcing OUT3 Forcing OUT4	(T) (H) (Z) (E)	0...wxyz	
		C931	R/S	INT	Forcing OUT5	(T)	0...x000	
		C940	R/S	INT	Forcing do1 Forcing do2 Forcing do3 Forcing do4	(T) (H) (Z) (E)	0...wxyz	
		C941	R/S	INT	Forcing do5 Forcing do6	(T) (H)	0...wx00	

Extended operating level					Parameter entry at extended operating level		(function no: 3)
Code		**Descr.**	**R/W**	**Type**	**Description**	**Range**	**Rem.**
B2		Entry1	R/W	INT	Parameter identification number	0 ... 32767	**N**
		Entry2	R/W	INT	Parameter identification number	0 ... 32767	
		Entry3	R/W	INT	Parameter identification number	0 ... 32767	
		...					
		Entry11	R/W	INT	Parameter identification number	0 ... 32767	
		Entry12	R/W	INT	Parameter identification number	0 ... 32767	
		Hold	R/W	INT		0 ... 13	**O**

Write field bus data					Parameter entry for field bus data		(function no.: 4)
Code		**Descr.**	**R/W**	**Type**	**Description**	**Range**	**Rem.**
B2		Entry1	R/W	INT	Parameter identification number	0 ... 32767	**N**
		Entry2	R/W	INT	Parameter identification number	0 ... 32767	
		Entry3	R/W	INT	Parameter identification number	0 ... 32767	
		...					
		Entry11	R/W	INT	Parameter identification number	0 ... 32767	
		Entry12	R/W	INT	Parameter identification number	0 ... 32767	

Read field bus data					Parameter entry for field bus data		(function no.: 5)
Code		**Descr.**	**R/W**	**Type**	**Description**	**Range**	**Rem.**
B2		Entry1	R/W	INT	Parameter identification number	0 ... 32767	**N**
		Entry2	R/W	INT	Parameter identification number	0 ... 32767	
		Entry3	R/W	INT	Parameter identification number	0 ... 32767	
		...					
		Entry 11	R/W	INT	Parameter identification number	0 ... 32767	
		Entry12	R/W	INT	Parameter identification number	0 ... 32767	

Rem N ***Entry 1 ... 12***
Value = 0 means ‘unused entry’

Rem O ***Hold***
Value = 0 means ‘Hold on main operating page’
Value = 1 means ‘Hold on status display at extended operating level’
Value = 2 ... 13 means ‘ Hold on entry 1 ... 12’

From firmware version 3.3 (October 1997), parameter setting at extended operating level can be done via the interface. The parameters are checked by the interface so that only valid parameters can be marked. However, note that a valid, marked parameter may not be displayed, because it is not displayed by the operation in the actual controller configuration.
Example: `LimL1` can be marked via interface, however, it is displayed only, if alarm 1 is configured.
The written parameters are effective immediately. After writing, a change to the main operating page, and to the entry marked Hold is made automatically. The entry identification is given in the following table.

Set-point

Code	Parameter
257	w0
258	w100
259	w2
260	Grw+
261	Grw-
262	Grw2
263	LC-
264	LC+

Limit 1

Code	Parameter
513	LimL1
514	LimH1
515	Lxsd1

Limit 2

Code	Parameter
769	LimL2
770	LimH2
771	Lxsd2

Limit 3

Code	Parameter
1025	LimL3
1026	LimH3
1027	Lxsd3

Limit 4

Code	Parameter
1281	LimL4
1282	LimH4
1283	Lxsd4

Tuning

Code	Parameter
1537	YOptm
1538	dYopt
1539	POpt
1540	Oxsd
1541	Trig1
1542	Trig2

Tuning

Code	Parameter
1543	Trig3
1544	ORes1
1545	ORes2
1546	Tu1
1547	Vmax1
1548	Tu2
1549	Vmax2

Addit. param.

Code	Parameter
1793	Xsh
1794	Tpuls
1795	Tm
1796	Xsd1
1797	LW
1798	Xsd2
1799	Xsh1
1800	Xsh2
1801	Y2
1802	Ymin
1803	Ymax
1804	Y0
1805	ParNr
1806	ParNr (read-only)

Valid parameter

Code	Parameter
2049	Xp1
2050	Xp2
2051	Tn1
2052	Tv1
2053	T1
2054	T2

Parameter set 0

Code	Parameter
2305	Xp1 0
2306	Xp2 0
2307	Tn1 0
2308	Tv1 0
2309	T1 0

Parameter set 0

Code	Parameter
2310	T2 0

Parameter set 1

Code	Parameter
2561	Xp1 1
2562	Xp2 1
2563	Tn1 1
2564	Tv1 1
2565	T1 1
2566	T2 1

Parameter set 2

Code	Parameter
2817	Xp1 2
2818	Xp2 2
2919	Tn1 2
2820	Tv1 2
2821	T1 2
2822	T2 2

Parameter set 3

Code	Parameter
3073	Xp1 3
3074	Xp2 3
3075	Tn1 3
3076	Tv1 3
3077	T1 3
3078	T2 3

Process value

Code	Parameter
3329	Tdz
3330	N0
3331	a
3332	b

Signl. process. INP 1

Code	Parameter
3585	X1in
3586	X1out
3587	X2in
3588	X2out
3589	m
3590	b
3591	gain
3592	Tf

Signl. process. INP 3

Code	Parameter
3841	m 3
3842	b 3
3843	gain 3
3844	Tf 3

Signl. process. INP 4

Code	Parameter
4097	m 4
4098	b 4
4099	gain 4
4100	Tf 4

Signl. process. INP 5

Code	Parameter
4353	m 5
4354	b 5
4355	gain 5
4356	Tf 5

Signl. process. INP 6

Code	Parameter
4609	m 6
4610	b 6
4611	gain 6
4612	Tf 6

Miscellaneous

Code	Parameter
4865	FKey
4866	Blck1
4867	Blck2

Timer

Code	Parameter
5121	TS.Y

Timer	
Code	**Parameter**
5122	TS.MD
5123	TS.HM
5124	TE.Y
5125	TE.MD
5126	TE.HM

Analog prog. Recp 1	
Code	**Parameter**
6401	Wmode
6402	Pmode
6403	Pnext
6404	LC-
6405	LC+
6406	Wp0
6407	——
6408	Tp1
6409	Wp1
6410	Tp2
6411	Wp2
...	...
6446	Tp20
6447	Wp20

Digital prog. Recp 1	
Code	**Parameter**
6657	D0
6658	——
6659	Td1
6660	D1
6661	Td2
6662	D2
...	...
6697	Td20
6698	D20

Analog prog. Recp 2	
Code	**Parameter**
6913	Wmode
6914	Pmode
6915	Pnext
6916	LC-
6917	LC+
6918	Wp0
6919	——
6920	Tp1
6921	Wp1
6922	Tp2
6923	Wp2
...	...
6958	Tp20
6959	Wp20

Digital prog. Recp 2	
Code	**Parameter**
7169	D0
7170	——
7171	Td1
7172	D1
7173	Td2
7174	D2
...	...
7209	Td20
7210	D20

Analog prog. Recp 3	
Code	**Parameter**
7425	Wmode
7426	Pmode
7427	Pnext
7428	LC-
7429	LC+
7430	Wp0
7431	——
7432	Tp1
7433	Wp1
7434	Tp2
7435	Wp2
...	...
7470	Tp20
7471	Wp20

Digital prog. Recp 2	
Code	**Parameter**
7681	D0
7682	——
7683	Td1
7684	D1
7685	Td2
7686	D2
...	...
7721	Td20
7722	D20

Signals	
Code	**Parameter**
7937	——
7938	——
7939	——
7940	Clock
5125	TE.MD
5126	TE.HM

Set-point signals	
Code	**Parameter**
8193	Wint
8194	Wext
8195	dWext
8196	dW (Dec. point = 1)
8197	dW (dec. point = 2)
8198	Wsel
8199	Weff

Controller signals	
Code	**Parameter**
8449	Y
8450	Yp
8451	xw
8452	X1
8453	X2
8454	X3
8455	z
8456	OVC
8457	Xeff

Input signals	
Code	**Parameter**
8705	INP1
8706	INP1r
8707	INP3
8708	INP3r
8709	INP4
8710	INP4r
8711	INP5
8712	INP5r
8713	INP6
8714	INP6r

Prog. signals	
Code	**Parameter**
8961	Wp
8962	tBrut
8963	tNet
8964	tRest
8965	PNr

Rapid recovery	
Code	**Parameter**
9217	XwOnY
9218	XwOnX
9219	GrwOn

Calibration INP1	
Code	**Parameter**
9473	x0c (PT100)
9474	x0c
9475	x100c

Calibration INP6	
Code	**Parameter**
9729	x0c
9730	x100c

Other	
Code	**Parameter**
9985	Status 1
9986	Finp1
9987	Finp3
9988	Finp4
9989	Finp5

Other	
Code	**Parameter**
9990	Finp6
9991	Fdi
9992	Fout1
9993	Fout3
9994	Fdo
9995	XFail
9996	Status di1
9997	Status di2
9998	Clock Hour
9999	Clock Minute
10000	Clock Day
10001	Clock Month
10002	Clock Year
10003	Clock Weekday
10004	Contr Status 1
10005	Contr Status 2
10006	Contr Status 3
10007	Y/ Y2
10008	PI/P
10009	A/M
10010	OStart
10011	We/Wi
10012	W/W2
10013	W/dW
10014	Coff
10015	WState
10016	Wnvol
10017	Wvol
10018	Tune Status
10019	ParNeff
10020	Prog Status 1
10021	Prog Status 2
10022	Wend
10023	Seg_AD
10024	PRun
10025	PRset
10026	Pset
10027	Alarm Status 1
10028	Yman

7.2.3 INPUT (FB no.: 61 Type no.: 110)

All data concerning the measurement and processing of all (analog/digital) input values are grouped in function block 'INPUT'

Processdata

General	Input value processing of analog, digital signals (function no.: 0)					
Code	**Descr.**	**R/W**	**Type**	**Description**	**Range**	**Rem.**
00	Block	R	Block	Block access (1, 3, 5...8)		
1	Input_x_Fail	R	ST1	Signal input x fail		**A**
3	INP1	R	FP	Signal input 1		
5	INP3	R	FP	Signal input 3		
6	INP4	R	FP	Signal input 4		
7	INP5	R	FP	Signal input 5		
8	INP6	R	FP	Signal input 6		
10	Block	R	Block	Block access (13...18)		
11	State_di1	R	ST1	Digital inputs di1...di6		**B**
12	State_di2	R	ST1	Digital inputs di7...di12		**C**
13	INP1A[1)]	R	FP	Signal input 1 physical value		
14	INP3A[1)]	R	FP	Signal input 3 physical value		
15	INP4A[1)]	R	FP	Signal input 4 physical value		
16	INP5A[1)]	R	FP	Signal input 5 physical value		
17	INP6A[1)]	R	FP	Signal input 6 physical value		
18	Function Type	R	INT	Type no. of function block	110	

Rem A ***Status byte Input_X_Fail:***

MSB							LSB
D7	D6	**D5**	**D4**	**D3**	**D2**	D1	**D0**

Bit no.	Name	Allocation	Status '0'	Status '1'
D0	INP1F	Input 1 fail	no	yes
D1	'0'	Always '0'		
D2	INP3F	Input 3 fail	no	yes
D3	INP4F	Input 4 fail	no	yes
D4	INP5F	Input 5 fail	no	yes
D5	INP6F	Input 6 fail	no	yes
D6	'1'	Always '1'		
D7		Parity		

Rem B ***State_di1 (digital inputs di1 ... di6)***

MSB							LSB
D7	D6	D5	D4	D3	D2	D1	D0

Bit no.	Name	Allocation	Status '0'	Status '1'
D0	di1	Digital input 1	off	on
D1	di2	Digital input 2	off	on
D2	di3	Digital input 3	off	on
D3	di4	Digital input 4	off	on
D4	di5	Digital input 5	off	on
D5	di6	Digital input 6	off	on
D6	'1'	Always '1'		
D7		Parity		

Rem C ***State_di2 (digital inputs di7 ... di12)***

MSB							LSB
D7	D6	D5	D4	D3	D2	D1	D0

Bit no.	Name	Allocation	Status '0'	Status '1'
D0	di7	Digital input 7	off	on
D1	di8	Digital input 8	off	on
D2	di9	Digital input 9	off	on
D3	di10	Digital input 10	off	on
D4	di11	Digital input 11	off	on
D5	di12	Digital input 12	off	on
D6	'1'	Always '1'		
D7		Parity		

1) Values before signal processing stage or measured value correction.

ME/V1 Input value INP1 : measurement and processing (function no.: 1)

Code	Descr.	R/W	Type	Description	Range	Rem.
31	X0c	R/W	INT	Trigger f. calibration X0	0..1	
32	X100c	R/W	INT	Trigger f. calibration X100	0..1	

ME/V6 Input value INP6 : measurement and processing (function no.: 11)

Code	Descr.	R/W	Type	Description	Range	Rem.
31	X0c	R/W	INT	Trigger f. calibration X0	0..1	
32	X100c	R/W	INT	Trigger f. calibration X100	0..1	

Timer Timer Funktion [1] (Funktionsnr: 13)

Code	Descr.	R/W	Type	Description	Range	Rem.
01	State_Clock	R	ST1	Status 1		D
30	Block	R	Block	Block access (Code 31...36)		
31	ClkH [2]	R/W	INT	Time hours	0...23	
32	ClkMi [2]	R/W	INT	Time minutes	0...59	
33	ClkD [2]	R/W	INT	Time day	1...31	
34	ClkMt [2]	R/W	INT	Time month	1...12	
35	ClkY [2,3]	R/W	INT	Time year	70...169	
36	ClkDW [2,4]	R/W	INT	Time weekday	0...6	

Rem D ***Status byte State_Clock***

MSB							LSB
D7	D6	D5	D4	D3	**D2**	**D1**	**D0**

Bit no.	Name	Allocation	Status '0'	Status '1'
D0	ClkEr	Clock Error	no	yes
D1	T1En	Timer1 enabled		
D2	T1Out	Timer1 status		
D3...D5	'0'	Always '0'		
D6	'1'	Always '1'		
D7		Parity		

T1Out	T1En	
0	0	Timer not active
0	1	enabled
1	0	Output active
1	1	not defined

Parameter- a. configurationdata

General Analog, digital signal input processing (function no.: 0)

Code		Descr.	R/W	Type	Description		Range	Rem.
B3		C180	R/W	INT	S X2: signal source for S2	(T)	0..wxyz	
					SWext: signal source for W_{ext}	(H)		
					S dW: signal source for dW	(Z)		
					S z: signal source for z	(E)		
		C190	R/W	INT	SWi/e: signal source for W_{int}/W_{ext}	(T)	0..wxyz	
					STrac: signal source for WTrac	(H)		
					SWdon: signal source for dw on/off	(Z)		
					Sw/W2: signal source for w/w2	(E)		
		C191	R/W	INT	S A/M: signal source for auto/manual	(T)	0..wxyz	
					SPI/P: signal source for FB on/off	(H)		
					SY2on: signal source for Y2	(Z)		
					SCoff: signal source for controller off	(E)		
		C192	R/W	INT	Prog: signal source for start/stop	(T)	0.x000	

1) Only valid with module B with real time clock, e.g. RS485 (d.c. for PROFIBUS and INTERBUS)
2) When reading, the actual internally available time is specified. During specification, order 'year-month-day-hour-minute' for correct checking must be met.
3) Calculation of the actual year: data range 70...169, corresponds to 1970...2069; example: value 96 corresponds to year 1996, value 101 corresponds to year 2001.
4) Signification: 0=Monday (first weekday), 1=Tuesday,...6=Sunday; Values are freely adjustable and are not evaluated presently.

ME/V1					Measured value INP1 : measurement and processsing			(function no.:1)
Code		**Descr.**	**R/W**	**Type**	**Description**		Range	Rem
B2		$X1_{in}$	R/W	FP	Measured value correction X1 input		-999...9999	
		$X1_{out}$	R/W	FP	Measured value correction X1 output		-999...9999	
		$X2_{in}$	R/W	FP	Measured value correction X2 input		-999...9999	
		$X2_{out}$	R/W	FP	Measured value correction X2 output		-999...9999	
B3		X0	R/W	FP	Phys. value at 0%		-999...9999	
		X100	R/W	FP	Phys. value at 100%		-999...9999	
		XFail	R/W	FP	Substitute value with sensor failure		-999...9999	
		Tfm	R/W	FP	Filter time constants measurement value processing		0.0...999.9	
		Tkref	R/W	INT	Specified TC		0...60°C	
		C200	R/W	INT	Type: sensor type Unit: unit Dp: decimal point	(T,H) (Z) (E)	0..xxyz	
		C205	R/W	INT	Fail: sensor break behaviour Stk: source TC Xkorr: enable process value correction	(T) (H) (Z)	1..wxy0	

Sv1					Signal processing level for INP1			(function no.: 2)
Code		**Descr.**	**R/W**	**Type**	**Description**		Range	Rem.
B2		m	R/W	FP	Scaling: gradient m		0..999.9	
		b	R/W	FP	Scaling: offset b		-999..9999	
		gain	R/W	FP	Square root extraction: gain		0 .. 9.999	
		Tf	R/W	FP	Pre-processing: filter time constant		0.0 .. 999.9	
B3		xs1	R/W	FP	Segment point 1: X value		-999..9999	
		ys1	R/W	FP	Segment point 1: Y value		-999..9999	
		xs2	R/W	FP	Segment point 2: X value		-999..9999	
		ys2	R/W	FP	Segment point 2: Y value		-999..9999	
		xs3	R/W	FP	Segment point 3: X value		-999..9999	
		ys3	R/W	FP	Segment point 3: Y value		-999..9999	
		xs4	R/W	FP	Segment point 4: X value		-999..9999[1)]	
		ys4	R/W	FP	Segment point 4: Y value		-999..9999	
		xs5	R/W	FP	Segment point 5: X value		-999..9999[1)]	
		ys5	R/W	FP	Segment point 5: Y value		-999..9999	
		xs6	R/W	FP	Segment point 6: X value		-999..9999[1)]	
		ys6	R/W	FP	Segment point 6: Y value		-999..9999	
		xs7	R/W	FP	Segment point 7: X value		-999..9999[1)]	
		ys7	R/W	FP	Segment point 7: Y value		-999..9999	
		xs8	R/W	FP	Segment point 8: X value		-999..9999[1)]	
		ys8	R/W	FP	Segment point 8: Y value		-999..9999	
		C220	R/W	FP	Func1: function selection 1 Func2: function selection 2 LDP: decimal point	(T) (H) (E)	0...wx0z	

The functions for measurement value processing and measurement of inputs INP3, INP4, INP5, INP6 are structured identically. INP3 and INP4 are available only with options card C fitted.

ME/Vx					Measurement value INPx: measurement and processing			(function no.: 5, 7, 9, 11)
Code		**Descr.**	**R/W**	**Type**	**Description**		Range	**Rem.**
B3		X0	R/W	FP	Phys. value at 0%		-999..9999	
		X100	R/W	FP	Phys. value at 100%		-999..9999	
		XFail	R/W	FP	Substitute value at sensor fail		-999..9999	
		Tfm	R/W	FP	Filter time constant Input value processing		0.0 .. 999.9	
		INPx (Cxx0)	R/W	INT	Type: sensor type Dp: decimal point	(T,H) (E)	0..xx0y	
		Option (Cx05)	R/W	INT	Fail: sensor break behaviour	(T)	0..x000	

1) Datum has switch-off function; additional data value '-32000'

The functions for signal pre-processing of inputs INP3, INP4, INP5, INP6 are structured identically. INP3 and INP4 are only available with options card C fitted.

Svx					Signal processing stage for INPx			(function no.: 6,8,10,12)
Code		**Descr.**	**R/W**	**Type**	**Description**		Range	Rem.
B2		m	R/W	FP	Scaling: gradient m		0 .. 999.9	1)
		b	R/W	FP	Scaling: offset b		-999..9999	
		gain	R/W	FP	Square root extraction: gain		0 .. 9.999	
		Tf	R/W	FP	Pre-processing: filter time constant		0.0 .. 999.9	
B3		xs1	R/W	FP	Segment point 1 : X value		-999..9999	2)3)
		ys1	R/W	FP	Segment point 1: Y value		-999..9999	
		xs2	R/W	FP	Segment point 2: X value		-999..9999	
		ys2	R/W	FP	Segment point 2: Y value		-999..9999	
		xs3	R/W	FP	Segment point 3: X value		-999.. 9999	
		ys3	R/W	FP	Segment point 3: Y value		-999.. 9999	
		xs4	R/W	FP	Segment point 4: X value		-999.. 9999	
		ys4	R/W	FP	Segment point 4: Y value		-999.. 9999	
		xs5	R/W	FP	Segment point 5: X value		-999.. 9999	
		ys5	R/W	FP	Segment point 5: Y value		-999.. 9999	
		xs6	R/W	FP	Segment point 6: X value		-999.. 9999	
		ys6	R/W	FP	Segment point 6: Y value		-999.. 9999	
		xs7	R/W	FP	Segment point 7: X value		-999.. 9999	
		ys7	R/W	FP	Segment point 7: Y value		-999.. 9999	
		xs8	R/W	FP	Segment point 8: X value		-999.. 9999	
		ys8	R/W	FP	Segment point 8: Y value		-999.. 9999	
		Cxx0	R/W	INT	Func1: function selection 1 Func2: function selection 2 NSeg: number segments LDP: decimal point	(T) (H) (Z) (E)	0..wxyz	1)2)

Timer					Timer Function [4]			(function no.: 13)
Code		**Descr.**	**R/W**	**Type**	**Description**		**Range**	**Rem.**
B2		T1SY	R/W	INT	Timer 1 start value year		0...255	
		T1SMt	R/W	INT	Timer 1 start value month		1..12	
		T1SD	R/W	INT	Timer 1 start value day		1 .. 31	
		T1SH	R/W	INT	Timer 1 start value hours		0 .. 23	
		T1SMi	R/W	INT	Timer 1 start value minutes		0 .. 59	
		T1EY	R/W	INT	Timer 1 end value year		0...255	5)
		T1EMt	R/W	INT	Timer 1 end value month		1..12	
		T1ED	R/W	INT	Timer 1 end value days		1 .. 31	
		T1EH	R/W	INT	Timer 1 end value hours		0 .. 23	
		T1EMi	R/W	INT	Timer 1 end value minutes		0 .. 59	
B3		C905	R/W	INT	TmMd: Operating mode timer 1	(T)	0..x000	

1) Defined only for
KS92: for input 5 and 6 (function 10, 12)
KS94: for Input 3, 4 (if module option C provided), 5 and 6 (function 6, 8, 10 and 12) .
2) Only defined, if the relevant SW option was enabled.
3) Defined only for KS94, INP4, provided that module option C is fitted.
4) Defined only for module B with real time clock (d.c. for PROFIBUS and INTERBUS).
5) Calculation of the actual year: Data range 70...169, corresponds to 1970...2069; example: value 96 corresponds to year 1996, value 101 corresponds to year 2001.

7.2.4 CONTR (FB no.: 50 Type no.: 90)

All data concerning the controller are grouped in function block 'CONTR'.

Processdata

General (function no.: 0)

Code	Descr.	R/W	Type	Description	Range	Rem
00	Block	R	Block	Block access (1...9)		
1	Status 1	R	ST1	Status 1		**A**
2	Status 2	R	ST1	Status 2		**B**
3	W	R	FP	Eff. set-point		
4	X	R	FP	Eff. process value		
5	Y	R	FP	Effective correcting variable		
6	xw	R	FP	Control deviation		
7	x1	R	FP	Variable 1		
8	x2	R	FP	Auxiliary variable 2		
9	x3	R	FP	Auxiliary variable 3		
10	Block	R	Block	Block access (11, 13...16, 18)		
11	Status 3	R	ST1	Status 3		**C**
13	Yp	R	FP	Position feedback		
14	z	R	FP	Auxiliary variable		
15	OVC+	R	FP	Override control +		
16	OVC-	R	FP	Override control -		
18	Type	R	INT	Type no of function block	90	
20	Block	R	Block	Block access (21...23)		
21	Wext	R	FP	Ext. set-point		
22	dW_extern	R	FP	Set-point offset		
23	Wlim	R	FP	Set-point limiting W min/max		
30	Block	R	Block	Block access (31...38)		
31	y/Y2	R/W	INT	Additional correcting value on/off	0..1	
32	PI/P	R/W	INT	PI/P switch-over	0..1	
33	A/M	R/W	INT	Manual/automatic switch-over	0..1	
34	OStart	R/W	INT	Self-tuning start	0..1	
35	We/i	R/W	INT	Wext/Wint switch-over	0..1	
36	w/W2	R/W	INT	w/W2 switch-over	0..1	
37	w/dW	R/W	INT	Corrective set-point off/on	0..1	
38	Coff	R/W	INT	Controller on/off	0..1	

Rem A ***Status1: (Code 01)***

MSB							LSB
D7	D6	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**

Bit no.	Name	Allocation	Status '0'	Status '1'
D0	Y1	Switching output	off	on
D1	Y2	Switching output	off	on
D2	A/M	Auto/manual	auto	manual
D3	y/Y2	y/Y2 switch-over	y	Y2
D4	Coff	Controller switched off	no	yes
D5	XFail	Sensor fail	no	yes
D6	'1'	Always '1'		
D7		Parity		

Rem B ***Status2: (Code 02)***

MSB							LSB
D7	D6	**D5**	**D4**	D3	D2	**D1**	**D0**

Bit no.	Name	Allocation	Status '0'	Status '1'
D0	GRW	Gradient function	not active	active
D1	BAND	Bandwidth control	not active	active
D2	RCV	Rapid recovery	no	yes
D3	'0'	Always '0'		
D4	PI/P	PI/P status	PI	P
D5	CFail	Controller status	Ok	not Ok
D6	'1'	Always '1'		
D7		Parity		

Rem C ***Status 3: (Code 11)***

MSB							LSB
D7	D6	D5	D4	D3	**D2**	**D1**	**D0**

Bit no.	Name	Allocation	Status '0'	Status '1'
D0	Xtrk	Int. set-point with X	off	on
D1	DOVC-	Override control- with 3-pnt. stepping controller	off	on
D2	DOVC+	Override control+ with 3-pnt. stepping controller	off	on
D3...D5	'0'	Always '0'		
D6	'1'	Always '1'		
D7		Parity		

Set-point — Set-point processing (Function no.:1)

Code	Descr.	R/W	Type	Description	Range	Rem.
1	WState	R	ST1	Set-point status		D
30	Block	R	Block	Block access (31...33)		
31	Wnvol	R/W	FP	Int. set-point, non volatile	-999..9999	
32	Wvol	R/W	FP	Int. set-point, volatile	-999..9999	
33	WdW	R/W	FP	Offset set-point / corrective value	-999..9999	

Rem D ***WState: (Code 01)***

MSB							LSB
D7	D6	D5	**D4**	**D3**	**D2**	**D1**	**D0**

Bit no.	Name	Allocation	Status '0'	Status '1'
D0	w/W2	w/W2 switch-over	w	W2
D1	We/Wi	Wext/Wint		
D2	Wp/Wi	Wprog/Wint		
D3	w/dW	Set-point correction active		
D4	w/dWe	Ext. set-point active		
D5	'0'	Always '0'		
D6	'1'	Always '1'		
D7		Parity		

wp/wi	we/wi	Set-point source
0	1	External
1	0	Programmer
1	1	Internal

w/dwe	w/dw	Set-point correction
0	0	Not active
0	1	Active, internal
1	0	Active, external

Correcting variable — Correcting variable processing (function no.:4)

Code	Descr.	R/W	Type	Description	Range	Rem.
30	Block	R	Block	Block access (31, 32)		
31	dYman	R/W	FP	Difference correcting variable	-210..210	
32	Yman	R/W	FP	Absolute correcting variable	-105..105	
33	DAC ®	R/W	FP	Start of automatic calibration of position feedback Yp	0...1	

Tuning				Self-tuning		(function no.:5)
Code	**Descr.**	**R/W**	**Type**	**Description**	**Range**	**Rem.**
00	Block	R	Block	Block access (1, 3)		
1	State_Tune1	R	ST1	Status Tuning		**E**
3	ParNeff	R	INT	Eff. parameter set number	0...3	
30	Block	R	Block	Block access (31...39)		
31	ParNr	R/W	INT	Parameter set number effective	1 .. 4	
32	Tu1	R	FP	Delay time heating	0...9999	
33	Vmax1	R	FP	Rate of increase heating	0,000...9,999	
34	Kp1	R	FP	Process gain heating	0,000...9,999	
35	MSG1	R	INT	Error code of heating self-tuning	0...8	
36	Tu2	R	FP	Delay time cooling	0...9999	
37	Vmax2	R	FP	Rate of increase cooling	0,000...9,999	
38	Kp2	R	FP	Process gain cooling	0,000...9,999	
39	MSG2	R	INT	Error code of self-tuning cooling	0...8	

Rem E ***Status 1 Tuning 'State_Tune1'***

MSB							LSB
D7	D6	D5	D4	D3	**D2**	**D1**	**D0**

Bit no.	**Name**	**Allocation**	**Status '0'**	**Status '1'**
D0	OStab	Process at rest	no	yes
D1	Orun	Self-tuning operation	off	on
D2	Oerr	Self-tuning result	Ok	Error
D3...D5	'0'	Always '0'		
D6	'1'	Always '1'		
D7		Parity		

Programmer				Programmer processing		(function no.: 10)
Code	**Desc.**	**R/W**	**Type**	**Description**	**Range**	**Rem.**
00	Block	R	Block	Block access (1...9)		
1	State_Prog1	R	ST1	Status 1 programmerr		**F**
2	State_Prog2	R	ST1	Status 2 programmer		**G**
3	PNreff	R	FP	Eff. program number	1..3	
4	Tnet	R	FP	Program time net	0 .. 9999	
5	Tbrut	R	FP	Program time gross	0 .. 9999	
6	Wp	R	FP	Programmer set-point	-999 .. 9999	
7	Trest	R	FP	Programmer rest time	0 .. 9999	
8	Wend	R	FP	Act. segment end value	-999 .. 9999	
9	Seg AD	R	INT	Segment no. analog/digital	0101 .. 2020	
30	Block	R	Block	Block access (31...37)		
31	Pnr	R/W	INT	Program number effective	1 .. 3	
32	PRun	R/W	INT	Program stop/run	0 .. 1	
33	PRset	R/W	INT	Program continue/reset	0 .. 1	
34	PSearch*	R/W	INT	Start program research	0 .. 1	
35	PSet	R/S	FP	Program preset value Pmode = Seg Pmode = Time	1...20 0...9999 (min)	
36	LC-	R/W	FP	Bandwidth lower limit	0...9999 a. '____'	
37	LC+	L/S	FP	Bandwidth upper limit	0...9999 a. '____'	1)

* Function is presently not supported.

1) LC-/LC+ are used as set-point gradient. These data are effective only, if no programmer is configured although a software option for programmer is provided. These data are identical with LC-/LC+ of the first program.

Rem F ***Status 1 programmer 'State_Prog1'***

MSB							LSB
D7	D6	D5	D4	D3	**D2**	**D1**	**D0**

Bit no.	Name	Allocation	Status '0'	Status '1'
D0	PRun	Progr. run	stop	run
D1	PEnd	Progr. end	no	yes
D2	PRes	Progr. reset	off	on
D3...D5	'0'	Always '0'		
D6	'1'	Always '1'		
D7		Parity		

Rem G ***Status 2 programmer 'State_Prog2'***

MSB							LSB
D7	D6	D5	D4	**D3**	**D2**	**D1**	**D0**

Bit no.	Name	Allocation	Status '0'	Status '1'
D0	Sp1	Output 1	off	on
D1	Sp2	Output 2	off	on
D2	Sp3	Output 3	off	on
D3	Sp4	Output 4	off	on
D3	'0'	Always '0'		
D6	'1'	Always '1'		
D7		Parity		

Parameter a. configurationdata

General (function no.: 0)

Code		Descr.	R/W	Type	Description		Range	Rem.
B2		Xwonx	R/W	BCD	Deviation value for X tracking		0 .. 9999	
		Xwony	R/W	BCD	Deviation value for Y tracking		0 .. 9999	
		Grwon	R/W	BCD	Gradient for reaching the target value		0,01 .. 99,99 /min	
B3		C103(Xn0)	R/W	FP	Lower calibration limit X1		-999..9999	
		C104(Xn100)	R/W	FP	Upper calibration limit X1		-999..9999	
		C108(X_{min})	R/W	FP	Lower process value limit		-999..9999	
		C109(X_{max})	R/W	FP	Upper process value limit		-999..9999	
		C110	R/W	FP	S factor		0.01 .. 99.99	
		C100	R/W	INT	CFunc: controller function CType: controller type WFunc: set-point function	(T,H) (Z) (E)	0..xxyz	
		C101	R/W	INT	CMode: controller output action CDiff: x, Xw-differentiation CFail: behaviour with sensor failure	(T) (H) (Z)	0..xyz0	
		C102	R/W	INT	XnDp: Decimal point for f. X standard	(E)	0..000x	
		C105	R/W	INT	C Aux: Auxiliary variable C OVC: Output limiting	(T,H) (Z)	0.. xxy0	
		C106	R/W	INT	WTrac: Tracking int. set-point WD: Set-point offset WSel: Set-point selection	(T) (H) (Z)	0..xyz0	
		C107	R/W	INT	Ratio: Ratio control function XDp: Decimal point for process value	(T) (E)	0..x00y	
		C700	R/W	INT	OMode: Type of self-tuning OCond: Process at rest OCntr: Operating mode contr. tuning	(T) (H) (E)	0..xy0z	

Set-point					Set-point processing		(function no.: 1)
Code		**Descr.**	**R/W**	**Type**	**Description**	**Range**	**Rem.**
B2		W0	R/W	FP	Lower set-point limit f. Weff	-999..9999	
		W100	R/W	FP	Upper set-point limit f. Weff	-999..9999	
		W2	R/W	FP	Additional set-point	-999..9999	
		Grw+	R/W	FP	Set-point gradient plus	>0..9.999	1)
		Grw-	R/W	FP	Set-point gradient minus	>0..9.999	
		Grw2	R/W	FP	Set-point gradient W2	>0..9.999	

Process value					Process value processing		(function no.: 2)
Code		**Descr.**	**R/W**	**Type**	**Description**	**Range**	**Rem.**
B2		N0	R/W	FP	Zero offset/ratio	-999..9999	
		a	R/W	FP	Factor a / 3-element control	-9.99..99.99	
		b	R/W	FP	Factor b / mean value control	0..9.999	
		Tdz	R/W	FP	Differentiation time constant for auxiliary variable	0...9999	

Algo					Control algorithm		(function no: 3)
Code		**Descr.**	**R/W**	**Type**	**Description**	**Range**	**Rem.**
B2		Xsh	R/W	FP	Neutral zone	0.2 .. 999.9%	
		Tpuls	R/W	FP	Min. pulse length	0.1..999.9%	1)
		Tm	R/W	FP	Motor actuator travel time	10..9999s	
		Xsd1	R/W	FP	Switching difference signaller	0..9999 %	
		LW	R/W	FP	Trigger point separation additional contact	-999..9999	
		Xsd2	R/W	FP	Switching difference additional contact	0..9999 %	
		Xsh1	R/W	FP	Neutral zone	0.0 .. 999.9%	
		Xsh2	R/W	FP	Neutral zone	0.0 .. 999.9 %	
		Xp1	R/W	FP	Act. proportional band 1	0.1..999.9	
		Tn1	R/W	FP	Act. integral action time 1	0..9999	
		Tv1	R/W	FP	Act. derivative action time 1	0..9999	
		T1	R/W	FP	Act. min. cycle time 1	0.4..999.9	
		Xp2	R/W	FP	Act. proportional band 2	0.1..999.9	
		Tn2	R/W	FP	Act. integral action time 2	0..9999	
		Tv2	R/W	FP	Act. derivative action time 2	0..9999	
		T2	R/W	FP	Act. min. cycle time 2	0.4..999.9	

Correcting variable					Correcting variable processing		(function no.: 4)
Code		**Descr.**	**R/W**	**Type**	**Description**	**Range**	**Rem.**
B2		Y_{min}	R/W	FP	Lower correcting variable limiting	-105..105	
		Y_{max}	R/W	FP	Upper correcting variable limiting	-105..105	
		Y2	R/W	FP	Additional correcting value	-105..105	
		Y0	R/W	FP	Working point for correcting variable	-105..105	

Tuning					Self-tuning		(function no: 5)
Code		**Descr.**	**R/W**	**Type**	**Description**	**Range**	**Rem.**
B2		YOptm	R/W	FP	Correcting variable during process at rest	-105..105	
		dYopt	R/W	FP	Step height with identification	5..100	
		OXsd	R/W	FP	Hysteresis with parameter set switch-over	0.0..9999	
		Trig1 2)	R/W	FP	Switch-over point 1	0.0..9999	
		Trig2 2)	R/W	FP	Switch-over point 2	0.0..9999	
		Trig3 2)	R/W	FP	Switch-over point 3	0.0..9999	
		POpt	R/W	INT	Parameter set which shall be optimized	0...3	

1) Datum has switch-off function; additional data value '-32000'.
2) The user must ensure that condition Trig1 < Trig2 < Trig3 is met for trigger points.

Paramset x				Control parameter set 1...4		(function no.: 6,7,8,9)
Code	**Descr.**	**R/W**	**Type**	**Description**	**Range**	**Rem.**
B2	Xp1	R/W	FP	Proportional band 1	0.1..999.9	
	Tn1	R/W	FP	Integral action time 1	0..9999	
	Tv1	R/W	FP	Derivative action time 1	0..9999	
	T1	R/W	FP	Min. cycle time 1	0.4..999.9	
	Xp2	R/W	FP	Proportional band 2	0.1..999.9	
	Tn2	R/W	FP	Integral action time 2	0..9999	
	Tv2	R/W	FP	Derivative action time 2	0..9999	
	T2	R/W	FP	Min. cycle time 2	0.4..999.9	

Programmer				Programmer operation		(function no.: 10)
Code	**Descr.**	**R/W**	**Type**	**Description**	Range	**Rem.**
B3	C120	R/W	INT	PSel: Source for program selection (T)	0...1	
				PwrUp:Behaviour with mains recovery (H)	0...4	
				Pend: Behaviour with program end (Z)	0...3	
				PStrt: Function with Run/Stop (E)	0...1	

The values defined for a program (max. 20 segment points and 4 control outputs) are distributed to 4 functions because of the limited transmission buffer.

Programm x 1				Programmer parameter set part 1		(function no.: 11,15,19)
Code	**Descr.**	**R/W**	**Type**	**Description**	**Range**	**Rem.**
B2	Wp1	R/W	FP	W value segment	0-999..9999	
	Wp2	R/W	FP	W value segment 1	-999..9999	
	Wp3	R/W	FP	W value segment 2	-999..9999	
	Wp4	R/W	FP	W value segment 3	-999..9999	
	Wp5	R/W	FP	W value segment 4	-999..9999	
	Wp6	R/W	FP	W value segment 5	-999..9999	
	Wp7	R/W	FP	W value segment 6	-999..9999	
	Wp8	R/W	FP	W value segment 7	-999..9999	
	Wp9	R/W	FP	W value segment 8	-999..9999	
	Wp10	R/W	FP	W value segment 9	-999..9999	
	Wp0	R/W	FP	Reset value W0	-999..9999	
	LC-	R/W	INT	Bandwidth lower limit	0..9999	
	LC+[1)]	R/W	INT	Bandwidth upper limit	0..9999	
	TP1[1)]	R/W	INT	Segment time analog	0..9999	
	TP2[1)]	R/W	INT	T value segment 1	0..9999	
	TP3[1)]	R/W	INT	T value segment 2	0..9999	
	TP4[1)]	R/W	INT	T value segment 3	0..9999	
	TP5[1)]	R/W	INT	T value segment 4	0..9999	
	TP6[1)]	R/W	INT	T value segment 5	0..9999	
	TP7[1)]	R/W	INT	T value segment 6	0..9999	
	TP8[1)]	R/W	INT	T value segment 7	0..9999	
	TP9[1)]	R/W	INT	T value segment 8	0..9999	
	TP10[1)]	R/W	INT	T value segment 9	0..9999	
	Wmode	R/W	INT	Change mode	0..1	
	Pmode	R/W	INT	Preset mode (0=segment; 1=time)	0..1	
	PNext[1)]	R/W	INT	Follow-up program	1..3	

1) Datum has switch-off function; additional data value '-32000'.

Programm x 2				Programmer parameter set part 2		(function no.: 12,16,20)
Code	**Descr.**	**R/W**	**Type**	**Description**	**Range**	**Rem.**
B2	Wp11	R/W	FP	W value segment 10	-999..9999	
	Wp12	R/W	FP	W value segment 11	-999..9999	
	Wp13	R/W	FP	W value segment 12	-999..9999	
	Wp14	R/W	FP	W value segment 13	-999..9999	
	Wp15	R/W	FP	W value segment 14	-999..9999	
	Wp16	R/W	FP	W value segment 15	-999..9999	
	Wp17	R/W	FP	W value segment 16	-999..9999	
	Wp18	R/W	FP	W value segment 17	-999..9999	
	Wp19	R/W	FP	W value segment 18	-999..9999	
	Wp20	R/W	FP	W value segment 19	-999..9999	
	Tp11 [1)]	R/W	INT	T value segment 10	0...9999	
	Tp12 [1)]	R/W	INT	T value segment 11	0...9999	
	Tp13 [1)]	R/W	INT	T value segment 12	0...9999	
	Tp14 [1)]	R/W	INT	T value segment 13	0...9999	
	Tp15 [1)]	R/W	INT	T value segment 14	0...9999	
	Tp16 [1)]	R/W	INT	T value segment 15	0...9999	
	Tp17 [1)]	R/W	INT	T value segment 16	0...9999	
	Tp18 [1)]	R/W	INT	T value segment 17	0...9999	
	Tp19 [1)]	R/W	INT	T value segment 18	0...9999	
	Tp20 [1)]	R/W	INT	T value segment 19	0...9999	

Programm x 3				Programmer parameter set part 3		(function no.: 13,17,21)
Code	**Descr.**	**R/W**	**Type**	**Description**	**Range**	**Rem.**
B2	Td1 [1)]	R/W	INT	T value segment	00..9999	
	Td2 [1)]	R/S	INT	T value segment 1	0..9999	
	Td3 [1)]	R/W	INT	T value segment 2	0..9999	
	Td4 [1)]	R/W	INT	T value segment 3	0..9999	
	Td5 [1)]	R/W	INT	T value segment 4	0..9999	
	Td6 [1)]	R/W	INT	T value segment 5	0..9999	
	Td7 [1)]	R/W	INT	T value segment 6	0..9999	
	Td8 [1)]	R/W	INT	T value segment 7	0..9999	
	Td9 [1)]	R/W	INT	T value segment 8	0..9999	
	Td10 [1)]	R/W	INT	T value segment 9	0..9999	
	D1	R/W	INT	Control output 1..4	0000 .. 1111	
	D2	R/W	INT	Control output 1..4	0000 .. 1111	
	D3	R/W	INT	Control output 1..4	0000 .. 1111	
	D4	R/W	INT	Control output 1..4	0000 .. 1111	
	D5	R/W	INT	Control output 1..4	0000 .. 1111	
	D6	R/W	INT	Control output 1..4	0000 .. 1111	
	D7	R/W	INT	Control output 1..4	0000 .. 1111	
	D8	R/W	INT	Control output 1..4	0000 .. 1111	
	D9	R/W	INT	Control output 1..4	0000 .. 1111	
	D10	R/W	INT	Control output 1..4	0000 .. 1111	
	D0	R/W	INT	Reset value control outputs 1..4	0000 .. 1111	

1) Datum has switch-off function; additional data value '-32000'.

Programm x 4 — Programmer parameter set part 4 (function no.: 14,18,22)

Code		Descr.	R/W	Type	Description	Range	Rem.
B2		Td11 [1]	R/W	INT	T value segment	0..9999	
		Td12 [1]	R/S	INT	T value segment 1	0..9999	
		Td13 [1]	R/W	INT	T value segment 2	0..9999	
		Td14 [1]	R/W	INT	T value segment 13	0..9999	
		Td15 [1]	R/W	INT	T value segment 14	0..9999	
		Td16 [1]	R/W	INT	T value segment 15	0..9999	
		Td17 [1]	R/W	INT	T value segment 16	0..9999	
		Td18 [1]	R/W	INT	T value segment 17	0..9999	
		Td19 [1]	R/W	INT	T value segment 18	0..9999	
		Td20 [1]	R/W	INT	T value segment 19	0..9999	
		D11	R/W	INT	Control output 1..4	0000 .. 1111	
		D12	R/W	INT	Control output 1..4	0000 .. 1111	
		D13	R/W	INT	Control output 1..4	0000 .. 1111	
		D14	R/W	INT	Control output 1..4	0000 .. 1111	
		D15	R/W	INT	Control output 1..4	0000 .. 1111	
		D16	R/W	INT	Control output 1..4	0000 .. 1111	
		D17	R/W	INT	Control output 1..4	0000 .. 1111	
		D18	R/W	INT	Control output 1..4	0000 .. 1111	
		D19	R/W	INT	Control output 1..4	0000 .. 1111	
		D20	R/W	INT	Control output 1..4	0000 .. 1111	

DAC ® — Actuator monitoring (Digital Actor Control) (function no.: 26)

Code		Descr.	R/W	Type	Description	Range	Rem.
B2		—-	R/W	FP	always ‘0’	0	
		—-	R/W	FP	always ‘0’	0	
		—-	R/W	FP	always ‘0’	0	
		—-	R/W	FP	always ‘0’	0	
		—-	R/W	FP	always ‘0’	0	
		DAC ®	R/W	INT	DAC ® function control	0...1	
		—-	R/W	INT	always ‘0’	0	
		—-	R/W	INT	always ‘0’	0	
		—-	R/W	INT	always ‘0’	0	
		—-	R/W	INT	always ‘0’	0	

1) Datum has switch-off function; additional data value ‘-32000’.

7.2.5 ALARM (FB no.: 51 type no.: 45)

Function block 'ALARM' defines the overall alarm processing of the relevant controller.

Processdata

General (function no.: 0)

Code	Descr.	R/W	Type	Description	Range	Rem.
1	Status 1	R	ST1	Alarm status x		A
18	Type	R	INT	Type no. of function block	45	

Rem A ***Status 1***

MSB							LSB
D7	D6	D5	D4	**D3**	**D2**	**D1**	**D0**

Bit no.	Name	Allocation	Status '0'	Status '1'
D0	AL1	Alarm 1	off	on
D1	Al2	Alarm 2	off	on
D2	Al3	Alarm 3	off	on
D3	Al4	Alarm 4	off	on
D4, D5	'0'	Always '0'		
D6	'1'	Always '1'		
D7		Parity		

Parameter a. configurationdata

General (function no.: 0)

Code	Descr.	R/W	Type	Description		Range	Rem.
B2	Lim_{L1} 1)	R/W	FP	Lower limit value 1		-999..9999	
	Lim_{H1} 1)	R/W	FP	Upper limit value 1		-999..9999	
	x_{sd1}	R/W	FP	Switching difference 1		0..9999	
	Lim_{L2} 1)	R/W	FP	Lower limit value 2		-999..9999	
	Lim_{H2} 1)	R/W	FP	Upper limit value 2		-999..9999	
	x_{sd2}	R/W	FP	Switching difference 2		0..9999	
	Lim_{L3} 1)	R/W	FP	Lower limit value 3		-999..9999	
	Lim_{H3} 1)	R/W	FP	Upper limit value 3		-999..9999	
	x_{sd3}	R/W	FP	Switching difference 3		0..9999	
	Lim_{L4} 1)	R/W	FP	Lower limit value 4		-999..9999	
	Lim_{H4} 1)	R/W	FP	Upper limit value 4		-999..9999	
	x_{sd4}	R/W	FP	Switching difference 4		0..9999	
B3	C600 (ALARM1)	R/W	INT	Src: signal source Fnc: function Mod: mode	(T,H) (Z) (E)	0..xxyz	
	C620 (ALARM2)	R/W	INT	Src: signal source Fnc: function Mod: mode	(T,H) (Z) (E)	0..xxyz	
	C640 (ALARM3)	R/W	INT	Src: signal source Fnc: function Mod: mode	(T,H) (Z) (E)	0..xxyz	
	C660 (ALARM4)	R/W	INT	Src: signal source Fnc: function Mod: mode	(T,H) (Z) (E)	0..xxyz	

1) Datum has switch-off function; additional data value '-32000'.

7.2.6 OUTPUT (FB no.: 81 Type no.: 111)

In function block 'OUTPUT', all data concerning the signal processing of all outputs (analog/digital) are grouped.

Processdata

General (function no.: 0)

Code	Descr.	Access	Type	Description	Range	Rem.
18	Type	R		Type no. of function block	111	

Parameter a. configurationdata

General (function no.: 0)

Code		Descr.	R/W	Type	Description		Range	Rem.
B3		C500 (OUT1)	R/W	INT	Src: output signal source Type: output stage type Mode: actuator output action	(T,H) (Z) (E)	0..xxyz	
		C530 (OUT2)	R/W	INT	Src: output signal source Type: output stage type Mode: actuator output action	(T,H) (Z) (E)	0..xxyz	
		C560 (OUT3)	R/W	INT	Src: output signal source Type: output stage type Mode: actuator output action	(T,H) (Z) (E)	0..xxyz	
		C590 (OUT4)	R/W	INT	Src: output signal source Type: output stage type Mode: actuator output action	(T,H) (Z) (E)	0.. xxyz	
		C591 (OUT5)	R/W	INT	Src: output signal source Type: output stage type Mode: actuator output action	(T,H) (Z) (E)	0.. xxyz	
		C596 (DO5)	R/W	INT	Src: output signal source Mode: output inversion	(T,H) (E)	0..xx0y	
		C597 (DO6)	R/W	INT	Src: output signal source Mode: output inversion	(T,H) (E)	0..xx0y	

SV 1 Signalprocessing for OUT 3 (function no.: 1)

Code		Descr.	R/W	Type	Description		Range	Rem.
B3		X0_Out3	R/W	FP	Reference value for output of %		999..9999	1)
		X100_Out3	R/W	FP	Reference value for output of 100%		999..9999	
		xs1	R/W	FP	Segment 1 : x value		-999..9999	1) 2)
		ys1	R/W	FP	Segment 1: Y-value		-999..9999	
		xs2	R/W	FP	Segment 2: X-value		-999..9999	
		ys2	R/W	FP	Segment 2: Y-value		-999..9999	
		xs3 3)	R/W	FP	Segment 3: X-value		-999.. 9999	
		ys3	R/W	FP	Segment 3: Y-value		-999.. 9999	
		xs4 3)	R/W	FP	Segment 4: X-value		-999.. 9999	
		ys4	R/W	FP	Segment 4: Y-value		-999.. 9999	
		xs5 3)	R/W	FP	Segment 5: X-value		-999.. 9999	
		ys5	R/W	FP	Segment 5: Y-value		-999.. 9999	
		xs6 3)	R/W	FP	Segment 6: X-value		-999.. 9999	
		ys6	R/W	FP	Segment 6: Y-value		-999.. 9999	
		xs7 3)	R/W	FP	Segment 7: X-value		-999.. 9999	
		ys7	R/W	FP	Segment 7: Y-value		-999.. 9999	
		xs8 3)	R/W	FP	Segment 8: X-value		-999.. 9999	
		ys8	R/W	FP	Segment 8: Y-value		-999.. 9999	
		C565	R/W	INT	Func: function selection. DP: decimal point	(T) (E)	0..x00y	1)

1) Only defined, if HW option C is configured.
2) Only defined for KS94 with the relevant HW option.
3) Datum has switch-off function; additional data value '-32000'.

SV 2					Signalprocessing for OUT 1			(function no.: 1)
Code		**Descr.**	**R/W**	**Type**	**Descr.**		**Range**	**Rem.**
B3		X0_Out1	R/W	FP	Reference value for output of %		999..9999	
		X100_Out1	R/W	FP	Reference value for output of 100%		999..9999	
		C.505	R/W	FP	Func: function selection. DP: decimal point	(T) (E)	-999..9999	

8 Annex

8.1 Legend of terms and abbreviations

CHAR5/16	Data type (→ see page 16)
COM PROFIBUS	Configuration tool (also COM ET200) make Siemens for PROFIBUS
FB	Abbr. for function block
Fkt	Abbr. for function
FP	Data type (→ see page 16)
ET	Abbr. for Engineering Tool
Funktion	A partial function of a function block seen from the interface
Funktionsblock	Self-contained processing unit
GSD-Datei	Geräte Stammdaten file
HW	Geräte Stammdaten file
ICMP	Data type (→ see page 16)
INT	Data type (→ see page 16)
INTERBUS	Standard communication protocol to EN 50254
ISO1745	Standard communication protocol ISO 1745, ASCII-based
PC-Schnittstelle	Front-panel interface at the controller for connecting an engineering tool
PCI	Process Control Instrument
PCI-Protokoll	ISO 1745-based protocol, implemented for PMA controllers
PROFIBUS-DP	Standard communication protocol to EN50170 vol.2 (DP: decentral periphery)
RS422	Standard 4-wire connection 4-wire connection, full duplex, (EIA RS 422); here: separate send/receive channels with max. 32 units
RS485	Standard 2-wire connection, half duplex, (EIA RS 485)
S5 / S7	Siemens AG controller series
Serielle Schnittstelle	Rear bussable controller interface
ST1	Data type (→ see page 16)
SW	Abbr. f. software
TTL	Module signal level
Typ	A function type number is allocated to each function block.

9 Index

Index

Baudrate 5
Begriffe 54
Cable definition 5
cable specification. 8
connection. 8
controller adjustment 11
controlwords 16
Data types 17
Datenstrukturierung 30
Datentypen. 17
diagnosis 8
fieldbusdata 12
Forcing. 10
Funktionsblocknummer 23
Initiate 22
INTERBUS 12
INTERBUS diagnosis. 8
Interface 8
laying cables 8
Local 7
measurement earth 8
Multiplex. 15
multiplexed transmission 15
Network 5
Objekte. 23
DECADE_AC 28
MEM_RD_PAR 24
MEM_WR_PAR 24
SINGLE_AC 25
TEXT_AC 27
operation)local). 7
Outputs can be locked 10
PCP-Dienste 20
Process 14
Process data 11
process data modules 14
Processdatamodules 14
screening 8
Statuswords 15

Subject to alterations without notice.
Bei Änderungen erfolgt keine Mitteilung.
Sus réserve de modification sans avis préalable

© PMA Prozeß- und Maschinen-Automation GmbH
Postfach 310 229, D - 34058 Kassel
Printed in Germany 9499 040 56911 (9908)

A4